

Technische Universität Ilmenau

Fakultät für Informatik und Automatisierung

Fachgebiet Rechnerarchitektur



## **Diplomarbeit**

zur Erlangung des akademischen Grades

## **Diplominformatiker**

# **Integrierter Entwurfsprozess für Leiterplatten mit FPGA und beispielhafte Realisierung**

---

angefertigt von: Rongtao Dai

geboren: am 14. Nov 1979 in Shanghai

Studiengang: Informatik

Matrikel: 2001

Matrikelnummer: 32450

Bearbeitungszeitraum: 24.08.2009 - 24.02.2010

verantwortlicher Hochschullehrer: Prof. Dr.-Ing. habil. Wolfgang Fengler

Betreuer: Dr.-Ing. Bernd Däne

Dipl.-Inf. Marcus Müller

Inventarisierungsnummer: 2010-02-24/019/IN01/2231

---

## **Danksagung**

An dieser Stelle möchte ich den Menschen danken, die mich beim Erstellen dieser Diplomarbeit begleitet haben. Ein besonderer Dank gilt den Herren Prof. Dr.-Ing. habil. Wolfgang Fengler, Dr.-Ing. Berand Däne und Dipl.-Inf. Marcus Müller für ihre hervorragende und keineswegs selbstverständliche Betreuung sowie die fruchtbaren Diskussionen im Rahmen meiner Diplomarbeit. Insbesondere möchte ich mich bei Dr.-Ing. Alexander Fleischer, Dipl.-Ing. Stefan Humbla, und Herr Oliver Brandel bedanken, die mir dabei viele Tipps und Vorschläge gegeben haben und immer ein offenes Ohr für mich hatten.

## Kurzfassung

An der Technischen Universität Ilmenau wird im Rahmen eines Sonderforschungsprojektes 622 eine Nanopositionier- und Nanomessmaschine (NPM) entwickelt und aufgebaut. Viele Mitarbeiter von verschiedenen Fachgebieten beschäftigten sich mit diesem Sonderforschungsprojekt. Aufgabe des Fachgebietes Rechnerarchitekturen ist dabei, hochleistungsfähige und verteilte Informationsverarbeitungssysteme im Teilbereich C1 zu entwerfen und zu realisieren.

Die Grundlage dieser Arbeit basierte auf einer Multi-DSP-Architektur mit Hochgeschwindigkeitsverbindungen. Ein solches System besteht aus mehreren miteinander kommunizierenden, digitalen Signalprozessormodulen (DSPs). Mehrere Digital Signalprozessoren übernehmen dabei die Messwertverarbeitung. Als vorläufige Version 0.1 arbeitet das Kompaktsystem mit 3 DSP Modulen C6713. Damit die DSPs miteinander mit Hochgeschwindigkeit kommunizieren können, werden noch verschiedene Kommunikationsprotokolle benötigt. Neben dem McBSP- basierten Kommunikationssystem mit Switch und dem McBSP- basierten Kommunikationssystem mit Bus wurde eine Ringstruktur Kommunikationsvariante zwischen den Kommunikationspartnern entwickelt. Dabei wurde der LVDS-Standard als physikalische Schnittstelle benutzt.

Diese Arbeit zielt auf eine Verbesserung und Weiterentwicklung des DSP-Kompaktsystems ab. Es soll in dieser Arbeit das Platinen Redesign des Multi DSP Kompaktsystems unter Berücksichtigung elektronischer Regelungen mittels Elektronik Designsoftware Altium Designer realisiert und die Änderungen an der EMIF Speicherschnittstelle eingearbeitet werden.

Ein anderer Fokus dieser Arbeit liegt dabei auf die Signalintegritätssimulation über die LVDS Kommunikationsleitungen auf das Basisboard. Dazu werden zuerst die theoretischen Grundlagen der Simulationsverfahren vorgestellt. An Hand der in dieser Arbeit beschriebenen Experimentfälle werden dann eventuelle Puls und Werteänderungen untersucht.

Die während dieser Arbeit ausprobierten Simulationen ermöglichen notwendige Spannungsverlaufsuntersuchungen vor der Endproduktion. Wichtig dabei sind das praktische Kennenlernen der Simulationsverfahren und die Vermittlung grundlegender Vorkenntnisse. Für die zukünftige Leiterplattenproduktion ist es notwendig, zuvor eine sinnvolle Simulation durchzuführen.

## **Hinweis zum Kopier- und Markenrecht**

Cyclone™ II und Quartus™ II sind eingetragenes Markenzeichen der Altera Corporation.

Spartan™-IIE ist eingetragenes Markenzeichen der Xilinx Inc.

Eagle™ Layout Editor ist eingetragenes Markenzeichen von CadSoft Computer GmbH

Altium™ Design ist eingetragenes Markenzeichen von Altium Limited

Die Namen anderer Firmen, Produktbezeichnungen und Markenzeichen, die nicht ausdrücklich als Markenzeichen oder eingetragene Markenzeichen der betreffenden Hersteller erwähnt wurden, sind dennoch entsprechend den gesetzlichen Vorschriften als Markenzeichen geschützt.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung.....</b>	<b>1</b>
1.1	Motivation und Zielstellung .....	1
1.2	Überblick .....	2
<b>2</b>	<b>Simulatoren und Simulationsverfahren.....</b>	<b>4</b>
2.1	Hintergrund der Schaltungssimulation .....	4
2.2	Simulation analoger Schaltungen .....	6
2.2.1	SPICE .....	6
2.2.2	PSPICE.....	8
2.2.3	XSPICE .....	9
2.3	Simulationsanalyse .....	10
2.3.1	Gleichstromanalyse .....	11
2.3.2	Transientenanalyse .....	12
2.3.3	Wechselstrom Kleinsignal Analyse .....	13
2.3.4	Weitere Analysearten .....	14
2.4	Simulation digitaler Schaltungen.....	14
2.4.1	VHDL.....	14
2.4.2	VHDL Entwurfssimulation .....	16
2.5	Mixed-Signal-Simulation .....	17
2.5.1	VHDL-AMS.....	18
2.5.2	Mixed-Signal-Simulation im Altium Designer .....	20
2.6	Board-Level-Simulation im Altium Designer .....	23
2.6.1	IBIS .....	24
2.6.2	BSDL.....	26
2.6.3	Signalreflexion .....	26
2.6.4	Übersprechen.....	28
2.6.5	Reflexions- und Übersprechenssimulation.....	28

2.7	FPGA-Level-Simulation in Altium Designer.....	30
<b>3</b>	<b>Die Grundlage der verwendeten Hardware .....</b>	<b>33</b>
3.1	Das DSP-Modul D.Module.6713 von D.SignT.....	33
3.1.1	Die Speicher .....	34
3.1.2	Die externe Busschnittstelle .....	35
3.2	Das Basisboard .....	36
3.3	Die Backplane.....	37
<b>4</b>	<b>Kommunikationssysteme des seriellen Multi-DSP-Systems .....</b>	<b>39</b>
4.1	LVDS Technik.....	40
<b>5</b>	<b>Altium Designer .....</b>	<b>42</b>
<b>6</b>	<b>Vorbereitung der Leiterplatten Verdrahtung.....</b>	<b>44</b>
6.1	Top-Level-Schematic .....	44
6.2	Design Rules (DR), Entwurfsregeln Einstellung.....	46
6.2.1	Mindestabstand zwischen allen Leitungen, Pads, Komponenten .....	46
6.2.2	Leitungsbreite für verschiedene Verdrahtungen .....	48
6.2.3	Routing Via Style .....	49
6.2.4	Matched Net Lengths Rules .....	50
6.3	Design Rule Checking (DRC), Entwurfsregelprüfung.....	50
6.4	Electrical Rule Checking (ERC), Elektrische Regelprüfung.....	53
6.5	Behebung von Komponenten-Positionsfehlern .....	55
6.6	Hinzufügen zusätzlicher Layer .....	60
6.7	Einsatz von Vias .....	61
<b>7</b>	<b>Altium Designer Routing Technologie .....</b>	<b>63</b>
7.1	Autorouter in Altium Designer.....	63
7.1.1	Probleme des Autorouting.....	63
7.1.2	Fazit des Autorouting .....	65
7.2	Interacitve Routing Funktion.....	66

7.2.1	Eingabe einer nicht vordefinierten Netzbreite .....	66
7.2.2	Smart Interactive Routing .....	67
7.3	Interactive Length Tuning .....	68
7.3.1	Netzkategorie.....	68
7.3.2	High Speed Rules .....	69
7.4	Interactive Length Tuning Routing Funktion .....	70
7.4.1	Betroffene Problem bei Interactive Length Tuning .....	72
7.5	Einarbeiten von Änderungen .....	75
7.6	Schlusswort für das PCB Layout.....	77
<b>8</b>	<b>Das Signalintegritätsexperiment.....</b>	<b>78</b>
8.1	LVDS Standard in Cyclone FPGA .....	79
8.2	Vorbereitungen vor der Signalintegritätsanalyse .....	80
8.2.1	Manuelles Hinzufügen des Signal-Integrität-Modell (IBIS Modell) .....	80
8.2.2	Bearbeiten der FPGA Schematic Pins.....	83
8.2.3	FPGA Signal Manager .....	84
8.2.4	Signal Integrität Design Regeln Einstellung .....	85
8.2.5	Bauteil Technologie .....	89
8.3	Signalintegritätsanalyse mit dem Demoprojekt.....	91
8.4	Das Ergebnis des Demoprojekts.....	93
8.5	Benötigte Designänderungen für die Signalintegritätsanalyse .....	95
8.6	Ergebnis der Experimente .....	99
<b>9</b>	<b>Zusammenfassung und Ausblick.....</b>	<b>109</b>
	<b>Literaturverzeichnis.....</b>	<b>111</b>
	<b>Abbildungsverzeichnis .....</b>	<b>115</b>
	<b>Anhang .....</b>	<b>117</b>
	<b>Verzeichnisstruktur der beigelegten DVD.....</b>	<b>118</b>

# 1 Einleitung

## Sonderforschungsbereich 622

An der Technischen Universität Ilmenau wird seit 2002 in dem Sonderforschungsbereich 622 „Nanopositionier- und Nanomessmaschinen“ [Son07], die von der Deutschen Forschungsgemeinschaft gefördert wird, geforscht. Nanopositionier- und Nanomessmaschinen sind für die nanometergenaue Positionierung und Messung von Nanooberflächen- und Nanostrukturnormalen, von mechanischen und optischen Präzisionsteilen sowie für die Materialanalyse zuständig. Eine unangefochtene Spitzenstellung und eine überaus erfolgreiche Entwicklung bescheinigten Experten dem Forscherteam, welches sich unter Leitung von Professor Gerd Jäger befand, für das SFB. Weltweit ist dieses Positioniergerät mit seinem Messvolumen von 25 mm x 25 mm x 5 mm und einer Positionierungsicherheit von unter 10 nm, das genaueste seiner Art. Das Fachgebiet Rechnerarchitekturen arbeitet daran, ein Mehrprozessorsystem bzw. ein Multi-DSP-System für die Anwendung in Nanopositionier- und Nanomessmaschinen in Teilprojktbereich C1 „Hochleistungsinformationsverarbeitung mit eingebetteten Systemen“ zu entwerfen. Es existieren funktionierende Prototypen der Mehrprozessor-Hardware, die von Dipl. -Inf. Falk Berger entwickelt wurden.

### 1.1 Motivation und Zielstellung

Teil dieser Diplomarbeit ist eine Verbesserung und Weiterentwicklung des Kompaktsystems mit Altium Designer auf Basis einer vorliegenden Studienjahrarbeit [Dai09]. In meiner Studienjahrarbeit wurden die Bauteile und Footprints, die nicht in der Bibliothek enthalten waren, einfach selbst erstellt. Hierbei konnte entweder ein Wizard zu Hilfe genommen oder alles selbst mit Hersteller-Datenblättern gezeichnet werden. Der wichtigste bei dieser Arbeit behandelte Prozess war der Routing Prozess im Altium Designer.

Hierbei wurde der Altium Designer Winter 09 Build 8.2.0.16457 für den Routing Prozess verwendet. Die spätere Signalintegritätsanalyse wurde aber mit der aktuellen Sommerversion 09 Service Pack 2 Build 9.2.0.18802 durchgeführt. Das Erstellen der Schaltpläne ist im Prinzip so wie bei jedem anderen Programm auch. Durch die Aufteilung größerer Schaltungen auf mehrere Schematics ist es leicht möglich, Projekte einzelnen Baugruppen direkt zuzuordnen, die dann auf einem Main Schematic zusammengeführt werden.

Zum Erstellen des Layouts stehen viele Regeleinstellungen zur Verfügungen. Außerdem besteht die Möglichkeit, eigene Regeln selbst zu definieren. Mit der richtigen Regeleinstellung



lassen sich nicht nur die Vorbereitungsarbeiten für weitere wichtige Routingprozesse erklären, sondern auch die Charakteristik des Layouts demonstrieren. Das Programm stellt verschiedene Routingmethoden sowohl automatisch als auch manuell zur Verfügung, mit denen man entweder das ganze Layout oder auch nur einzelne Leiterbahnen routen kann.

Ein weiteres Ziel dieser Arbeit ist die Recherche aller heutigen Simulationsverfahren und Methoden sowohl auf Leiterplattenebene als auch auf FPGA-Ebene. Die Realisierung einer Signalintegritätsanalyse auf einem ausgewählten differentiellen LVDS Leitungspaar ist ebenfalls Schwerpunkt dieser Arbeit. In diesem Rahmen wurde zunächst ein IBIS Modell von der Hersteller-Website heruntergeladen. Zur Implementierung der Signalintegritätsanalyse wurden zunächst drei verschiedenartiger Experimentalfälle ausgedacht und unumgängliche Entwurfsänderungen vorgenommen. Bei diesem Experiment handelt es sich um eine Signalreflexionsanalyse und Analyse des Übersprechens auf zwei benachbarten LVDS Leitungen, der zeitliche Verlust bei der Signalübertragung durch gebogene Leitung sowie die Spannungsänderung auf den FPGA Pins resultieren aus der geringeren Leitungsbreite und dem vergrößerten Abstand des Loopback Systems.

## 1.2 Überblick

**Kapitel 2** ist ein Recherchekapitel. Darin werden mehrere Simulationsverfahren und die entsprechenden Theorien besprochen. Speziell geht es um die Implementierung der Board-Level-Simulation, bzw. die Reflexionsanalyse und Analyse des Übersprechens, welche für spätere Experimenten verwendet werden.

**Kapitel 3** der Arbeit befasst sich mit den Grundlagen der verwendeten Hardware. Hierbei werden die Funktionalität sowie die Komponenten des DSP-Moduls, der D.Module.6713s, des Basisboards und der Backplane kurz abgehandelt.

**Kapitel 4** beschäftigt sich mit der Kurzbeschreibung der drei Kommunikationssysteme. Speziell geht es um die Beschreibung der LVDS Verbindung.

**Kapitel 5** beschreibt die verwendete Leiterplatten-Entwurfssoftware Altium Designer. Neben den besonderen Vorteilen dieser Entwurfssoftware werden die wesentlichen Prozesse der Leiterplattenherstellung anhand eines Ablaufdiagrammes gezeigt.

**Kapitel 6** behandelt die Vorbereitungsarbeiten zur Leiterplattenverdrahtung. Dazu gehören die Behebung der Komponenten-Positionsfehler, das Hinzufügen zusätzlicher Layer und die Durchkontaktierung. Das Kapitel bietet auch einen Überblick über die in Altium

Designer für unseren Entwurf verwendeten elektronischen Regeln sowie deren Festlegung und die Bedeutung der Parameter. Weiterhin wird auch die Entwurfsregelprüfung bzw. elektrische Regelprüfung behandelt. Dabei wird die Programmfunktion zur Auswahl und Nachprüfung der Design-Regeln-Typen vorgestellt.

**Kapitel 7** vermittelt die Routing Technologie in Altium Designer. Interactive Routing Funktion und Autorouting werden in diesem Kapitel miteinander verglichen und dabei untersucht, welche Unterschiede zwischen dem interaktiven Routing und dem Autorouting bestehen. Besonders wird die Interactive Length Tuning Routing Funktion vorgestellt, in der die Funktion des Verbindens mit gleicher Netzlänge realisiert wird. Abschließend werden noch die schon vorliegenden Änderungen auf der Leiterplatte eingearbeitet.

**Kapitel 8** geht mit Hilfe des Altium Designer Differenzial Pair Demoprojektes und der Herstellersdatenblätter auf die Signalreflexionsanalyse und Analyse des Übersprechens auf dem Basisboard an Hand dreier Betrachtungsfälle ein. Die Ergebnisse, die in dieser Arbeit erreicht wurden, werden ebenfalls dargestellt.

Im Anhang dieser Arbeit befinden sich verschiedenes ergänzendes Materialien sowie eine Begleit DVD, auf der sämtliche elektronischen Daten, die zu dieser Arbeit gehören, abgespeichert sind.

## 2 Simulatoren und Simulationsverfahren

Die Weiterentwicklung der IC<sup>1</sup> Technologie ermöglicht es, die gesamten komplexen elektronischen Systeme auf einem einzigen Chip zu realisieren. Ein solches System umfasst in der Regel das digitale Signal und die analoge Signalverarbeitung. Simulationen werden eingesetzt, um das spätere Verhalten eines Schaltungsentwurfs (IC Design) abzuschätzen. Ausgehend von der mathematischen Beschreibung der physikalischen Zusammenhänge erlauben sie eine weitgehende Approximation an die Realität.

Dieses Kapitel befasst sich mit den Verfahren und Methoden der analogen, signalintegrierten Schaltungssimulation, Mixed signalintegrierten Schaltungssimulation und VHDL<sup>2</sup> Entwurfssimulation für digitale Simulation sowie deren Anwendung im Altium Designer.

### 2.1 Hintergrund der Schaltungssimulation

Unter Schaltungssimulation versteht man die Erstellung eines Signallaufzeit-Modells in der Schaltung mit dem Ziel, die tatsächlichen Schaltungsfunktionen zu simulieren.

In den frühen siebziger Jahren wurden Simulationstechnologien zur Überprüfung integrierter Schaltungsverhältnisse vor der Herstellung verwendet. Durch Erstellung von Transistor Strom- und Spannungsvariablen wird das Verhalten der Schaltung simuliert. Man nennt dies eine Analog-Simulation oder Circuit-Level-Simulation. Frühere Simulatoren konnten nur ein paar hundert Transistoren-Schaltung simulieren. Mit der Entwicklung der mathematischen Methode ist der Simulator nun in der Lage, größere Schaltungen in Griff zu bekommen. Der bekannteste ist der von der Universität Berkeley entwickelt SPICE-Simulator. Er umfasst verschiedene Versionen. SPICE<sup>3</sup> 1G und SPICE 2G sind in der FORTRON Sprache geschrieben. SPICE3 ist in C Sprache geschrieben. SPICE 2G und SPICE3 werden derzeit in vielen kommerziellen Simulatoren verwendet.

Der SPICE-Simulator entwickelt die Schaltungsgleichungen mit modifizierten Knotenpunkten (Nodes) unter Verwendung integrierter Analysemethoden. Es bietet nicht-lineare DC<sup>4</sup> -

---

<sup>1</sup> Die englische Bezeichnung Integrated circuit für integrierte Schaltung

<sup>2</sup> Very High Speed Integrated Circuit Hardware Description Language

<sup>3</sup> Simulation Program with Integrated Circuits Emphasis

<sup>4</sup> Die englische Bezeichnung direct current für Gleichstrom

Analyse, nichtlineare transiente Analyse und lineare AC<sup>5</sup>-Analyse. Die Transientenanalyse ist die wichtigste, aber auch zeitaufwändigste Validierungsmethode. In der Regel wird zuerst die numerische Integrationsmethode verwendet, um nichtlineare Differentialgleichungen in eine Reihe von linearen algebraischen Gleichungen umzuformulieren. Dann wird das Gaußsche Eliminationsverfahren angewandt, um die linearen Gleichungen zu lösen. Diese linearen Gleichungen sind aber nur in der Integrationszeit gültig. Deshalb muss die Anwendung der Integrationsmethode mit dem Fortschreiten der Simulation wiederholt werden, um ein neues System linearer Gleichungen zu erhalten. Je schneller sich das Signal verändert, umso kleiner sollten die Integrationsschritte sein, damit die Integrationsmethode zu möglichst genauen Lösungen führt. Daher erfordert die Transientenanalyse einer Vielzahl mathematischer Operationen. Genauereres diesbezüglich folgt in Abschnitt 2.2.1. [Har08]

Mit der Zunahme der IC-Integration ist die Verwendung des Circuit-Level-Simulators zur Simulation des Verhaltens des gesamten IC Chips nicht mehr rentabel. Deshalb muss nach neuen Techniken und Methoden gesucht werden. Das IC-Design enthält meist nur die Digital Features, so dass der gesamte Chip als eine Sammlung miteinander verbundener Logik-Gatter modelliert werden kann. Das daraus resultierende Simulationsverfahren wird als Gate-Level-Simulation oder Logik-Simulation bezeichnet. Durch diese Annahme kann der Simulator Tausende von Transistorschaltungen behandeln, verliert dabei aber nur einen kleinen Anteil der Signalgenauigkeit. Der Ausgangszustand jedes Modells kann durch Boolesche Operationen einfach aus dem Eingangszustand abgeleitet werden. Die Kosten dieser Operationen sind deutlich niedriger als die der arithmetischen Operationen der Circuit-Level-Simulation. Der dem Ausgangszustand zugewiesene Wert bleibt unverändert, solange der Eingangszustand nicht verändert wird. Der Gate-Level-Simulator überprüft die Gate des Modelles nur in dem Moment, in welchem der Eingangszustand geändert wird. Dies bedeutet, dass nur eine begrenzte Anzahl gesonderter, diskreter Schritte zur Aktivierung notwendig sind, wohingegen die Circuit-Level-Simulation die Prüfung zu jeder Zeit erfordert (kontinuierlich, Zeit-Schritte). Darum ist der digitale Simulator mit diskreter Methode hundertmal schneller als der analoge Simulator.

Wegen der Annahme der MOS<sup>6</sup>-Technologie, ermöglicht die Voltage-Controlled-Schalter in der MOS-Logik-Gate-Modell zu realisieren. Die Knotenspannung des Kontrollanschlusses

---

<sup>5</sup> Alternating current, die Bezeichnung für Wechselstrom

<sup>6</sup> Metal Oxide Semiconductor, war ein Mikroprozessor- und Elektronikrechnerbauteile-Hersteller

entscheidet den Schaltzustand. Diese Methode ist weniger effizient als die Gate-Level-Simulation, aber immer noch effizienter als die Circuit-Level-Simulation. Allgemein wird sie als Switch-Level-Simulation bezeichnet. [Her02]

Mit der weiteren Erhöhung der IC-Design-Abstraktion wurde auch ein Simulator für eine höhere Abstraktionsebene entwickelt. Dieser wird normalerweise als ein Behavioral-Level-Simulator bezeichnet. Die Behavioral-Simulation unterstützt die Zusammensetzung und die sequentielle digitale Logik. Einige Simulatoren unterstützen auch analoge Funktionen.

Zusätzlich verfügen die digitalen Bauelemente über verschiedene Hochsprachen zur analytischen Beschreibung, z.B. VHDL Sprache (siehe Abschnitt 2.4.1).

## **2.2 Simulation analoger Schaltungen**

Der Entwurf einer Schaltung muss vor der Produktion durch geeignete Analyseverfahren auf Korrektheit überprüft werden. Besonders bei der Herstellung integrierter Schaltkreise können so enorme Entwicklungskosten gespart werden. Schaltungssimulation heißt auch Schaltungsmessen. Dabei werden alle bekannten Gegebenheiten überprüft. Zwei wesentliche Punkte sind dabei zu beachten: nämlich erstens muss man die Eigenschaften des zu prüfenden Objektes kennen, um eine richtige Messung durchführen zu können, und zweitens muss man, um möglichst korrekte Analyseergebnisse zu bekommen, auch die passende Simulationsmethode wählen. Beim Entwurf integrierter analoger Schaltungen wird die Simulation als Hauptprüfungswerkzeug verwendet. Hierbei wird die Analogsimulation im Wesentlichen auf der elektrischen Ebene, aber auch auf Makro- und Blockebene eingesetzt. Außerdem basiert die Simulation auf der numerischen Lösung von Gleichungssystemen. Zur Schaltungssimulation existieren derzeit mehrere numerische Verfahren, wie in [Ana05] erläutert.

### **2.2.1 SPICE**

Nach [Spi02] wurde Die SPICE Software für analoge Schaltungssimulationen im Jahre 1972 von der University of California, Berkeley, Computer Aided Design Team unter Verwendung der FORTRAN Sprache entwickelt. In erst Linie wurde SPICE-Software für groß angelegte Computer Aided Design von integrierten Schaltungen benutzt. Die praktische Version von SPICE: SPICE 2G wurde im Jahre 1975 veröffentlicht, jedoch war die Betriebsumgebung des Programmes nur für Mid-Range Computer geeignet. 1985 wurde SPICE von der University of California in Berkeley in C Sprache umgeschrieben und im Jahr 1988 zum United States National Industry Standard erklärt. In der Zwischenzeit gibt es eine Vielzahl kommerzieller

Schaltkreis-Simulations-Software. SPICE Simulations-Software mit SPICE Kern hat sich vielfach in der praktischen Arbeit bewährt, so dass SPICE zur wohl populärsten elektronischen Schaltkreis Simulations-Software geworden ist.

SPICE benutzt das Knotenpotentialverfahren<sup>7</sup>, um die Netzwerke zu berechnen. Das Prinzip der Knotenpotentialanalyse wird bei einem Netzwerk mit  $m$  Knoten (Nodes) gibt es  $m - 1$  unabhängige Knotengleichungen, darum wird  $m$  Knoten auf  $m - 1$  Knoten reduziert. Da bei einem Knoten keine Gleichung aufgestellt werden muss, wird es zum Bezugspotential (Masse) erklärt. Dieser Knoten kann beliebig gewählt werden. Die üblichen Knoten werden beispielsweise von 1 bis  $n$  nummeriert und von diesen führen zum Bezugspotential die Potentialdifferenzen  $V_1 - V_n$  (Knotenspannungen). Mit Hilfe der Kirchhoffschen Regeln kann das Gleichungssystem für die Maschen wie folgt aufgestellt werden: [Joe08]

$$\begin{pmatrix} G_{11} & G_{12} & \cdots & G_{1n} \\ G_{21} & G_{22} & \cdots & G_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ G_{n1} & G_{n2} & \cdots & G_{nn} \end{pmatrix} * \begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{pmatrix} = \begin{pmatrix} I_1 \\ I_2 \\ \vdots \\ I_n \end{pmatrix}$$

Ströme  $I_1 - I_n$  bedeutet in diesem Gleichungssystem zwar die Summe aller in die Knoten 1 bis  $n$  fließenden Ströme, und die Ströme mit negativem Wert sind die aus dem Knoten ausfließt.  $V_1 - V_n$  sind wie vorher genannten Knotenspannungen.  $G_{ii}$  sind der Summe Leitwerte alle Zweige, welche an Knoten  $i$  verbunden sind.  $G_{ij}$  sind der Leitwert zwischen den benachbarten Knoten  $i$  und  $j$ . wenn man beim Durchlaufen der Masche von  $i$  nach  $j$  dieselbe Richtung eingeschlagen hat, wird es mit negativ gezählt. Wenn keine direkte Verbindung zwischen zwei Knoten existiert, wird an dieser Stelle eine Null eingetragen.

Im Allgemeinen sind Schaltungssimulationen die Methode für die Beschreibung der Modelleigenschaften des Schaltplanes. Der Ablauf der Schaltungssimulation wurde erst durch die im Schaltplan verwendeten Komponenten über Differenzialgleichungssysteme modelliert und später mit Hilfe eines analogen Solvers unter Verwendung verschiedener Algorithmen (z.B. Newton-Raphson-Integration) gelöst. Die Schaltungssimulation mit SPICE verwendet dasselbe Prinzip.

<sup>7</sup> Das Knotenpotentialverfahren ist ein Verfahren zur Netzwerkanalyse in der Elektrotechnik. Mit diesen Verfahren lassen sich die Knotenpotentiale eines elektrischen Netzwerks aus linearen Bauelementen bestimmen.

*Es ist das algorithmische Finden von Näherungslösungen für die systembeschreibenden Differentialgleichungen. Deren Zusammenhang wird von der Schaltungstopologie bestimmt und mittels einer Netzliste, welche die Bauelemente und deren Verbindungen beschreibt, an den Simulator übergeben. Die Bauelemente werden durch Modelle beschrieben, die teils physikalisch orientiert sind, aber auch vollkommen abstrakt formuliert sein können. [Wik09a]*

Zahlreiche Hersteller stellen neben ihrem Produkt auch die passenden SPICE Modelle zur Verfügung. Diese werden auch als Simulationsmodelle bezeichnet. Der Simulator von Altium Designer arbeitet mit PC-Versionen von SPICE, worauf im nächsten Abschnitt näher eingegangen wird. Darüber hinaus ist eine ausführliche Beschreibung des Simulationswerkzeuges (Mix-Signal-Simulation) in Altium Designer in Abschnitt 2.5.2 enthalten.

### **2.2.2 PSPICE**

PSPICE<sup>8</sup> ist eine Umgebung für die umfangreiche Simulation von analogen, digitalen und Mixed-Signal Schaltungen. Es wurde von der amerikanischen Firma Microsim auf der Grundlage der SPICE 2G Version für den Personal-Computer-Bereich weiterentwickelt. Im Jahre 1998 fusionierte der bekannte kommerzielle EDA<sup>9</sup>-Software-Entwickler OrCAD<sup>10</sup> offiziell mit Microsim. OrCAD übernahm das PSPICE Produkt von Microsim in sein eigenes EDA-System. Die derzeit vom Unternehmen OrCAD veröffentlichte neue PSPICE-Version hat in drei wesentlichen Bereichen zu einer erheblichen Verbesserung im Vergleich zur traditionellen SPICE Software geführt: [Hei09]

1. Die Monte-Carlo-Analyse, Worst-Case-Analyse und die Optimierung von Design komplexer Schaltung charakteristischen Analyse, basiert auf der Analyse der analoge Schaltungstechnik DC, AC und Transiente Merkmale.
2. Das Simulieren nicht nur analoger Schaltungen, sondern auch digitaler Schaltungen, der Analog / Digital Mixed Schaltungen.

---

<sup>8</sup> PSPICE ist die PC-Version des legendären Simulationsprogramms SPICE.

<sup>9</sup> Electronic Design Automation (EDA) ist ein Teilgebiet des Computer Aided Design (CAD) bzw. des Computer Aided Engineering (CAE) oder der Entwurfsautomatisierung in der Elektronik und Mikroelektronik / Nanoelektronik.

<sup>10</sup> OrCAD ist ein von der Firma Cadence Design Systems vertriebenes Programmpaket, das im Bereich EDA zum Entwickeln von elektronischen Schaltungen eingesetzt wird.

3. Die Integration hat sich erheblich verbessert. Nach Abschluss der Schaltplanzeichnung können Schaltungen direkt simuliert und die Ergebnisse der Simulation leicht beobachtet und analysiert werden.

Obwohl die Anwendung von PSPICE immer umfangreicher wird, gibt es aber auch deutliche Mängel. SPICE-Software wurde ursprünglich vor allem für das elektronische Informations-Schaltungsdesign entwickelt, daher sind die Modelle nur für elektronische Geräte mit niedrigem Energieverbrauch geeignet, nicht aber für elektronische Schaltungen in Hochspannungs- und Hochleistungsgeräten. Hier kann PSPICE sogar zu falschen Simulationsergebnissen führen.

### **2.2.3 XSPICE**

SPICE 3 war ursprünglich Vorgänger von XSPICE. Es ist eine erweiterte und verbesserte Version des beliebten analogen Schaltungssimulations-Programms SPICE. Im Jahre 1985 wurde SPICE 3 komplett neu in C Sprache geschrieben. Es ermöglicht, Simulationsergebnisse grafisch anzuzeigen. XSPICE wurde 1992 am Georgia Tech Research Institute (GTRI) als Instrument für die Simulation von Schaltungen und Systemen auf verschiedenen Abstraktionsebenen entwickelt.

Die wichtigsten Neuerungen von XSPICE sind seine Erweiterbarkeit und digitale Zustände für digitale Simulation. XSPICE erlaubt es dem Benutzer, Datentypen und auch Modelle selbst zu definieren. XSPICE ist zudem ein Standard geblieben, der von den meisten Programmen im- und exportiert werden kann.

XSPICE präsentiert in Richtung Mixed-Signal-Simulation (siehe Abschnitt 2.5.2) ein 12-Zustands-Modell mit drei digitalen Wertklassen und vier digitalen Signal-Stärkeklassen für die digitale Simulation. Die Werte der digitalen Wertklassen schließen die Werte Null (0) und Eins (1) für logische Zustände 0 und 1 sowie einen zusätzlichen Wert (U) für undefinierte Zustände ein. Außerdem werden die digitalen Signale nach ihrer Stärke in vier Klassen eingeteilt, nämlich starke (s), resistive (r), hochohmige (z) und unbestimmte Stärke (u). Verdeutlicht wird dies durch die folgende vollständige Auflösungstabelle [Bli06]. Wenn zwei Signale sich auf derselben Leitung befinden, kann XSPICE nach dieser Auflösungstabelle die Entscheidung treffen, welcher Wert und welche Stärke vorrangig sein sollen.



		Signal 1											
		0 s	1 s	U s	0 r	1 r	U r	0 z	1 z	U z	0 u	1 u	U u
Signal 2	0 s	0 s	U s	U s	0 s	0 s	0 s	0 s	0 s	0 s	0 s	U s	U s
	1 s	U s	1 s	U s	1 s	1 s	1 s	1 s	1 s	1 s	U s	1 s	1 s
	U s	U s	U s	U s	U s	U s	U s	U s	U s	U s	U s	U s	U s
	0 r	0 s	1 s	U s	0 r	U r	U r	0 r	0 r	0 r	0 u	U u	U u
	1 r	0 s	1 s	U s	U r	1 r	U r	1 r	1 r	1 r	U u	1 u	U u
	U r	0 s	1 s	U s	U r	U r	U r	U r	U r	U r	1 u	U u	U u
	0 z	0 s	1 s	U s	0 r	1 r	U r	0 z	U z	U z	0 u	U u	U u
	1 z	0 s	1 s	U s	0 r	1 r	U r	U z	1 z	U z	U u	1 u	U u
	U z	0 s	1 s	U s	0 r	1 r	U r	U z	U z	U z	U u	U u	U u
	0 u	0 s	U s	U s	0 u	U u	U u	0 u	U u	U u	0 u	U u	U u
	1 u	U s	1 s	U s	U u	1 u	U u	U u	1 u	U u	U u	1 u	U u
	U u	U s	1 s	U s	U u	U u	U u	U u	U u	U u	U u	U u	U u

**Tabelle 1: Auflösetabelle für digitale Werte in XSPICE**

## XSPICE Simulator

Der XSPICE Simulator ist das Software-Programm, welches für die mathematische Simulation einer Schaltung zuständig ist. Es übernimmt den Input in Form von Befehlen und Schaltungsbeschreibungen und erzeugt eine Datenausgabe (z.B. Spannung, Strom, digitale Zustände und Wellenformen), welche das Schaltungsverhalten beschreibt (siehe [Flc92]).

## 2.3 Simulationsanalyse

Sowohl XSPICE als auch PSPICE arbeiten in der analogen Simulation mit der Knoten-Spannungs-Analyse und bieten somit eine Vielzahl von Analyseformen. Dabei unterscheidet man die Analysearten in zwei wesentliche Grundbereiche, die Großsignalanalyse und die Kleinsignalanalyse. Die Großsignalanalyse stellt ein echtes Abbild des Schaltungsverhaltens dar. Bei der Kleinsignalanalyse wird zuerst die Schaltung im Arbeitspunkt linearisiert. Dann werden sämtliche Analysearten mit dieser linearisierten Schaltung durchgeführt.

### 2.3.1 Gleichstromanalyse

Die Gleichstromanalyse wird auch Großsignalanalyse (DC-Analyse) genannt. Die DC-Analyse berechnet den statischen Schaltungsarbeitspunkt durch Ersatz einer aufgetrennten Kapazität und einer kurzgeschlossenen Induktivität in der Schaltung in der Weise, dass unter konstanten Anregungsbedingungen die statische Lösung der Schaltung bestimmt wird. Für den späten Wechselstrom der Kleinsignalanalyse, müssen in der Schaltung zuerst die DC-Analyse zur Bestimmung der Steilheit von Halbleiterbauelementen und anderen Kleinsignalparametern durchgeführt werden. Um die Lösung einer Schaltung mit der Transientenanalyse in der angegebenen Zeitspanne zu gelangen, muss in diesem Moment die Schaltungsgleichung als Differentialgleichung dargestellt werden. Anschließend um die Differentialgleichung zu lösen, müssen zuerst Strom und Spannungswert der Energiespeicher-Komponenten mittels DC-Analyse ausgerechnet werden. Daher ist die DC-Analyse die Startlösung für alle anderen Analysen.

Enthalten die charakteristischen Gleichungen einer Schaltung oder eine ihrer Komponenten nichtlineare Betragsfunktionen oder zeigen diese Komponenten ein nichtlineares Übertragungsverhalten zwischen Ein- und Ausgangsgrößen, so handelt es sich um eine nichtlineare Schaltung. Außer ein paar passiven Netzwerkkomponenten, die lineare Schaltungen sind, ist die überwiegende Mehrheit der Schaltungen nichtlinear.

Eine nichtlineare Schaltung besteht aus nichtlinearen Komponenten. Die wesentlichen nichtlinearen Komponenten sind z.B. Dioden, Bipolar Transistoren, MOS-FET-Halbleiter-Geräte, Magnetische Geräte, optoelektronische Bauelemente sowie nichtlineare Widerstände, Kondensatoren, Induktivitäten usw.

Die in linearer DC-Analyse aufgestellte Schaltungsgleichung ist eine lineare algebraische Gleichung und die in nichtlinearer DC-Analyse aufgestellte Gleichung ist eine nichtlineare algebraische Gleichung. Die Lösung eines nichtlinearen Gleichungssystems ist viel komplexer als die Lösung eines Linearen Gleichungssystems, weil zunächst die nichtlineare Gleichung in eine lineare algebraische Gleichung umgestellt werden muss. Der Ablauf der DC-Analyse entspricht dem des Lösen des nichtlinearen Gleichungssystems aus der Transientenanalyse, also dem Newton-Raphson Verfahren. [Ana05]

Die wichtigsten Funktionen der DC-Analyse:

- Arbeitspunktbestimmung: statische Lösung des nichtlinearen Gleichungssystems.
- Berechnung der Übertragungskurve: Kurvenverlauf der Ausgangsgrößen über den variierten Eingangsgrößen.
- Berechnung der Übertragungsfunktion: DC-Transfer, Variation einer Komponente des Eingangsvektors u. diskreter Schritte im festgelegten Wertebereich.

Gleichstromanalyse ist der Startpunkt von anderer Analyseart wie z.B. Transientenanalyse und Wechselstrom Kleinsignal Analyse, die wird gleich in folgenden Abschnitt behandelt.

### 2.3.2 Transientenanalyse

Laut [Hei09] ist die Transientenanalyse in elektronischen Schaltungen die Berechnung des Verhaltens einer Schaltung über einen angegebenen Zeitbereich. Parametrisiert durch Start- und Endzeit sowie die Zahl der Simulationszwischen-schritte berechnet sie das Systemverhalten (also alle Knotenspannungen und Ströme) bei den vorgegebenen Eingangsspannungen. Bei der Transientenanalyse wird die Schaltung durch eine Gruppe von gewöhnlichen Differentialgleichungen beschrieben. Gegenstand der Transientenanalyse ist die Lösung dieser Differentialgleichungen. Durch Anwendung der Laplace Transformationsmethode für lineare Schaltungen kann die Systemantwort im Zeitbereich am Ausgang durch die Übertragungsfunktion des Netzwerkes berechnet werden. Die Laplace Transformationsmethode gilt aber nur für eine einfache lineare Schaltung. Wird die Schaltung komplexer, dann ist auch die Übertragungsfunktion sehr schwer herauszufinden. Darüber hinaus sind die meisten praktischen Schaltungen nichtlineare Schaltungen. Aus diesem Grund ist die Transientenanalyse mit Hilfe numerischer Verfahren nur zur Lösung gewöhnlicher Differentialgleichungen geeignet. Dieser Analyseart gilt nicht nur für lineare Schaltungen, sondern auch für nichtlineare Schaltungen. Numerische Verfahren zur Lösung von Differentialgleichungen sind auch als numerische Integrationsverfahren bekannt.

Bei der Transientenanalyse kann die Schaltung durch folgende Differentialgleichungen erster Ordnung ausgedrückt werden: [Die92]

$$x' = f(x, t), \text{ wobei } t_0 \leq t \leq T$$

Seine Anfangsbedingung:

$$x(t_0) = x_0$$

in impliziter Form:

$$\begin{cases} F(x, x', t) = 0 \\ x(t_0) = x_0 \end{cases}, \text{ wobei } t_0 \leq t \leq T$$

Dabei ist  $x$  die unbekannte Variable Vektor der Schaltung.  $x'$  ist Vektor der ersten Ableitung von  $x$  nach der Zeit  $t$ .  $x_0$  ist der Startvektor.  $t$  ist die Zeitvariable im Zeitbereich  $[t_0, T]$ .

### 2.3.3 Wechselstrom Kleinsignal Analyse

Die AC-Analyse ist die Berechnung der Wechselstrom-Ausgangsvariablen im Frequenzbereich. Mittels AC-Analyse berechnet das Simulationsprogramm zunächst den DC Arbeitspunkt der Schaltung und linearisiert das Gleichungssystem für alle nichtlinearen Bauelemente im Schaltkreis. Mit Hilfe der Fouriertransformation werden dann linearisierte Gleichungen im Frequenzbereich angegeben. Das Resultat kann grafisch als Bodediagramm dargestellt werden. [Ana05]

$$\vec{f}(\vec{x}(t), \dot{\vec{x}}(t), \vec{u}(t)) = \vec{0}$$

Linearisierung des Gleichungssystems im gewünschten Arbeitspunkt:

$$G\vec{x}(t) + C \frac{d\vec{x}(t)}{dt} - \vec{u}(t) = \vec{0}$$

Fouriertransformation:

$$\begin{aligned} G\vec{X} + Cj\omega\vec{X} - \vec{U} &= \vec{0} \\ (G + j\omega C)\vec{X} &= \vec{U} \end{aligned}$$

Lösung im Frequenzbereich für sinusförmige Erregung eines Eingangs  $U$  in diskreten Schritten:

$$\text{Betrag: } |X| = \sqrt{X_{Rc}^2 + X_{Im}^2}, \text{ Phase: } \Phi = \arctan\left(\frac{X_{Im}^2}{X_{Rc}^2}\right)$$

Vergleicht man die Wechselstrom-Kleinsignal-Analyse mit einer Transientenanalyse, bei welcher der zeitliche Verlauf der Spannungen und Ströme ohne jegliche Vereinfachungen

berechnet wird, erkennt man, dass die Wechselstrom-Kleinsignal-Analyse wesentlich schneller ist.

### 2.3.4 Weitere Analysearten

Neben den drei oben genannten Analyseformen bieten heutige Analog-Simulatoren, wie PSPICE oder XSPICE, noch weitere Analysearten (z.B. Rauschanalyse, Empfindlichkeitsanalyse, Monte-Carlo-Analyse, Worst-Case-Analyse, Fourieranalyse usw.), wie in [Ana05] erläutert.

**Rauschanalyse:** Die Rauschanalyse bestimmt durch Ermittlung von Rauschwerten Widerstände und Halbleiter durch Anwendung von Rausch-Spektraldichte-Diagrammen. Das System kann die folgenden Rauscharten berechnen: Eingangsrauschen, Ausgangsrauschen und Rauschen elektrischer Bauelemente.

**Empfindlichkeitsanalyse:** Berechnung der Empfindlichkeit der Schaltung gegenüber Parameteränderung

**Monte-Carlo-Analyse:** Ein statistisches Berechnungsverfahren für die Überprüfung von Prozess- oder Bauelementeschwankungen der Schaltung.

**Worst-Case-Analyse:** Suchen nach der maximalen Abweichung von Nominalwert.

**Fourieranalyse:** Die Fourieranalyse ist eine Frequenzbereichsanalyse zur Berechnung des Gleichanteils und der Harmonische aus dem Ergebnis der Transientenanalyse. Mit diesem Analyseverfahren kann das Ausgangssignal im Frequenzbereich dargestellt werden.

## 2.4 Simulation digitaler Schaltungen

Es ist allgemein bekannt, dass die gesamte reale Welt analog ist und die Umwandlung analoger in digitale Signale nur zur Erleichterung und weiteren Verarbeitung der Signale dient. Im Bereich der Digitalschaltungen wird fast ausschließlich eine Hardware-Beschreibungssprache als Eingabewerkzeug zur Beschreibung der Schaltung verwendet. Derzeit existieren mehrere verschiedene derartige Sprachen. Die davon am meisten verwendeten Sprachen sind VHDL und Verilog.

### 2.4.1 VHDL

Laut [Per02] ist die VHDL-Sprache eine High-Level-Sprache für den Schaltungsentwurf. Standardisiert wurde sie in den späten 80er Jahren und ist mittlerweile sehr weit verbreitet.

Derzeit erhält VHDL als IEEE<sup>11</sup> Industrie-Standard-Hardware-Beschreibungssprache Unterstützung durch eine Reihe von EDA-Unternehmen. Im Bereich der Elektrotechnik hat sie sich zur General-Purpose Hardware-Beschreibungssprache entwickelt.

Zweck der Anwendung der VHDL-Sprache ist die Beschreibung der Struktur, des Verhaltens, der Funktion und der Schnittstellen der digitalen Systeme. Außerdem enthält VHDL eine Reihe von Statements mit Hardware-Eigenschaften. Stil und Syntax der VHDL-Sprache sind der allgemeinen High-Level-Programmiersprache sehr ähnlich.

Zielstellung von VHDL: [Ste06]

- Beschreibung des Verhaltens von Schaltungen auf algorithmischer Ebene
- Kompilieren der Verhaltensbeschreibung auf Gatter-/Registerebene und Erzeugung der Netzliste
- Beschreibung von Gatterebene und Netzliste
- Simulation von Digitalschaltungen
- Erzeugen von Testmustern und Überprüfung der Ergebnisse
- VHDL besitzt alle Merkmale einer universellen Programmiersprache

Der Schaltungsentwurfsprozess mit VHDL-Sprache benötigt folgende Schritte:

1. Schreiben der Entwurfs Quelldateien unter Verwendung eines Text-Editors
2. Kompilieren der Quelldateien mit Kompilier-Werkzeugen
3. Funktionale Simulation (Optionaler Schritt): Absicherung der Zuverlässigkeit des Entwurfs und Beweisen seiner Integrität.
4. Kombination: Umwandlung der Quelldateien aus der Entwurfssprache in der Schaltung. Das oberste Ziel dieser Schritte ist das Erzeugen der Netzliste für die Gate-Level-Schaltung.

---

<sup>11</sup> Das Institute of Electrical and Electronics Engineers (IEEE) ist ein weltweiter Berufsverband von Ingenieuren aus den Bereichen Elektrotechnik und Informatik.

5. Layout, Verkabelung: Layout und Verkabelung erfolgen unter Verwendung der in Schritt 4 generierten Netzlistenstruktur. Ferner wird eine Programmierungsdatei für die spätere Programmierung erzeugt.
6. Post-Simulation. Dieser Schritt ist vor allem auf den Entwurf nach Layout und Verkabelung gerichtet und hinterfragt, ob die Entwurfsanforderungen noch erfüllt sind.
7. Programmierung

VHDL ist eine dominierende und vielseitige Sprache. Die oben beschriebenen sieben Schritte stellen lediglich allgemeine Schaltungsentwicklungsschritte mit VHDL dar. Sie sind nur ein vages Beispiel dafür, was es heißt, ein gegebenes Problem zunächst zu erfassen und dessen Lösung schließlich in eine Schaltung zu programmieren.

## **2.4.2 VHDL Entwurfssimulation**

Im VHDL Design ist die Entwurfssimulation ein sehr wichtiger Teil der Arbeit. Die gesamte Simulation ist in funktionale Simulation und Timing Simulation untergeteilt. Die funktionale Simulation validiert die Korrektheit der logischen Funktion des Entwurfsmoduls. Mit der Anwendung der Timing Simulation wird das zeitliche Verhalten des Entwurfsmoduls untersucht. Ob bei funktionaler Simulation oder bei Timing Simulation: Es gibt nur zwei Arten von Simulationsverfahren, nämlich die interaktive Simulation und das Testbench-Simulationsverfahren. [Kes06]

### **Interaktive Simulation**

Die meisten VHDL-Entwurfswerkzeuge unterstützen die echtzeitliche interaktive Simulations-Operation. Bei der interaktiven Simulation können das Eingangssignal leicht vorgegeben, Ausführungszeit und Ausführungsweise der Simulation bestimmt und die Eingangs- und Ausgangs-Wellenformen beobachtet werden. Bei der Benutzung von VHDL-Entwurfswerkzeugen mit interaktiver Simulationsfunktion kann die Testbench-Programmierungsprozedur entfallen, wodurch die Entwurfseffizienz erheblich verbessert wird. Einige dieser Werkzeuge verfügen jedoch über keine interaktive Simulationsfunktion, sodass der Designer deren Testbench-Programme selbst schreiben muss.

### **Testbench**

Testbench heißt, einen Testsignal-Generator in VHDL-Sprache zu schreiben und den Ausgang zu überprüfen. Es besteht aus zwei Teilen: Der eine ist für den Test des erzeugten Ein-

gangssignals des Modelles zuständig, ein weiterer Teil dient der Überprüfung des Ausgangssignals des Modelles.

Die meiste VHDL-Simulationssoftware erfordert keine Programmierung der Ausgangsüberprüfung des Modells in Testbench. Sie braucht nur im Testbench Code das Signal für das Testobjekt zu verhängen, durch die Kompilierung, Verbindung und Simulation können Eingangs- und Ausgangswellenform erzeugt werden. Testbench-Programmierung erfordert eine erhebliche Menge an Arbeit und manchmal ist der Code des Testbenchs noch länger als das getestete Objekt.

## **2.5 Mixed-Signal-Simulation**

In der heutigen Zeit enthalten moderne Schaltungen oft eine Mischung aus analogen und digitalen Schaltungen. Für die effiziente und genaue Simulation solcher Schaltungen ist eine Mischung aus analogen und digitalen Simulationstechniken erforderlich. Im traditionellen Schaltungsentwurf sind digitale und analoge Teile auf dem Entwurf strikt getrennt. Ein analoger Simulator löst mittels numerischer Verfahren komplexe Differentialgleichungssysteme (siehe Darstellung im Kapitel 2.2.1), wohingegen digitale Simulatoren auf Grund diverser Besonderheiten digitaler Schaltungen (Zeit- und Wertdiskretisierung, eindeutiger Signalfluss) lediglich einfache Datenstrukturen zu diskreten Zeitpunkten auswerten (Ereignissteuerung). Wegen der unterschiedlichen Verfahren für Analog- und Digitalschaltungen treten bei der Simulation einer Mixed-Signal-Schaltung erhebliche Probleme auf. [Mix05]

Dabei liegt die besondere Herausforderung der gemischten Simulation (Mixed-Signal-Simulation) in den verschiedenen Auswertungsgeschwindigkeiten der beiden Schaltungsarten. Ziel der gemischten Simulation ist es, die gesamte Leistungsbreite des Digitalsimulators zu nutzen, ohne dass darunter die Präzision der analogen Schaltungsberechnung leidet.

### **Erweiterte Simulatorkerne [Ant03]**

Die auf diese Weise erweiterten analogen Simulatoren, wie z. B. PSPICE und XSPICE) verfügen über analoge Beschreibungen digitaler Elemente und können so Simulationsergebnisse ohne Präzisions- und Laufzeitverluste liefern. Anders sieht dies bei erweiterten Digitalsimulatoren aus. Hier müssen analoge Bauteile je nach Anwendung digital erzeugt werden.



## **Simulatorenkopplung**

Eine weitere Möglichkeit besteht in der losen Koppelung eines digitalen und eines analogen Simulators. Hierbei muss jeder Verbindungspunkt zwischen den beiden Simulatoren über eine Konverterstruktur geführt werden, welche digitale Zustände auf analogen Signalen und analoge Signale auf digitalen Zuständen abbildet (Wertkonvertierung).

Im Rahmen eines übergeordneten Prozesses, welcher den Datenaustausch kontrolliert, werden beide Simulatoren gekoppelt und synchronisiert. Dabei erfolgt eine Abstimmung der Zeitschritte in den Berechnungen (Zeitschrittsynchronisation). Zusätzlich kann hierbei eine einheitliche Netzlistenverwaltung eingefügt werden. Im Folgenden geht es um genau diesen Fall.

### **2.5.1 VHDL-AMS**

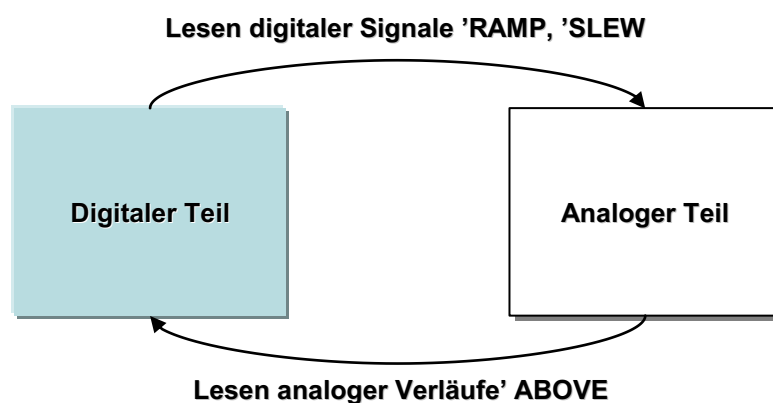
Traditionell werden digitale und analoge Schaltungen unter Verwendung unterschiedlicher Beschreibungssprachen, wie schon im vorherigen Kapitel behandelt, dargestellt und simuliert, nämlich analoge Schaltungen mit SPICE, digitale Schaltungen mit VHDL. Diese unabhängige Analyse-Methode für die Analyse und Überprüfung von Funktionalität und Leistung des gesamten Systems bringt viele Unannehmlichkeiten mit sich, wodurch es schwierig wird, aus dem gesamten System Analyse- und Simulationsergebnisse zu erhalten.

Zu diesem Zweck hat das IEEE in Jahr 1999 den IEEE Std 1076.1 Standard VHDL veröffentlicht, um die Beschreibungsfähigkeit für die VHDL analogen Schaltkreise und Mixed-Signal-Systeme zu erweitern. Der 1076.1 Standard, eine erweiterte Version des 1076 Standards, wurde später als VHDL-AMS bekannt und definiert. Dabei steht die Abkürzung AMS für Analog und Mixed-Signal.

Die wichtigsten Anwendungen von VHDL-AMS sind die Beschreibung des Verhaltens von Mixed-Signal-Geräten (z.B. A/D oder D/A Wandler) und Mixed-Signal-Systemen (siehe [Haa99]). Vor allem kann jede Art von SPICE-Geräte-Modellen mit dieser Sprache beschrieben werden. Obwohl die SPICE-Modelle zum Teil nicht zur VHDL 1076.1 Standard Sprache gehören, kann aber dennoch die VHDL-AMS-Sprache zur Beschreibung der verschiedenen Arten von SPICE Geräte-Modellen und Makro-Modellen genutzt werden. Dadurch wird der Anwendungsbereich der VHDL-AMS erweitert. Zweitens kann man ein VHDL-AMS-Modell in verschiedenen Sichtweisen betrachten. Es ermöglicht die Beschreibung des Verhaltens der Komponente direkt durch verschiedene Zuweisungen und Differentialgleichungen. Außerdem kann das physikalische Verhalten des Modells auch durch VHDL-AMS dargestellt werden.

## VHDL-AMS Simulation

Das wichtigste Ziel der VHDL-AMS-Anwendungen ist es, das gemischte analog-digitale Verhalten simulieren zu können. In einer VHDL-AMS-Beschreibung werden die analogen und digitalen Teilebeschreibungen sprachlich getrennt voneinander formuliert und eingeschätzt. Während der Auswertung des Analogteils werden vom Simulator digitale Signale als konstant angesehen. Entsprechend werden während der Auswertung des Digitalteils analoge Verläufe als konstant betrachtet. Laut [Haa03] sorgen Attribute für den Datenaustausch zwischen analogen und digitalen Werten (siehe Abbildung 2-1).



**Abbildung 2-1: Attribute zur Unterstützung des Wertaustausches zwischen dem digitalen und analogen Teil einer VHDL-AMS-Beschreibung**

Dabei kann der Signalverlauf mit Anstiegs- und Abfallzeit durch das Attribut **'Ramp** definiert werden. Das Attribut **'Slew** beschreibt Anstiegs- und Abfallsbegrenzung einer Quantity. **'Above** signalisiert Über- oder Unterschreiten einer Schwelle und erzeugt das digitale Ereignis.

Analog- und Digitalsimulator arbeiten auf unterschiedlichen Zeitachsen und mit unterschiedlichen Geschwindigkeiten, die aber später aufeinander synchronisiert werden müssen. Darum wird die Hilfe des Break-Statements benötigt.

Der VHDL-AMS Simulation besteht aus folgenden drei Phasen: [Sch05]

1. der Elaborationsphase. Bei dieser Phase handelt sich um den Aufbau des Simulationsmodells aus allen kompilierten VHDL-Modellen.
2. die Initialisierungsphase. In dieser Phase werden folgende Aufgaben erledigt: Zuweisung aller Default Werte, Ausführen aller digitalen Signalzuweisungsoperationen und einmalig-

ge Durchführung aller Prozesse. Bestimmen des Arbeitspunktes der analogen Komponenten nachher.

3. Der Exekutionsphase, Hier geht es um die Wiederholung der Berechnung aller Zuweisungen, Prozesse und des Differentialgleichungssystems bis zum Ende der Simulation.

Die VHDL-AMS-Sprache ist ähnlich der VHDL-Sprache und benutzt die Funktionen der VHDL-Sprache weiter. Die gewöhnlichen Differentialgleichungen werden verwendet, um das physikalische Modell zu beschreiben. Tatsächlich können alle physikalischen Prozesse, die sich durch gewöhnliche Differentialgleichungen erklären lassen, auch durch VHDL-AMS-Sprache beschrieben werden. Aus diesem Grund kann VHDL-AMS auch in vielen nicht elektrischen Bereichen zur Beschreibung des Systemverhaltens und zum Entwurf von Systemen verwendet werden.

## **2.5.2 Mixed-Signal-Simulation im Altium Designer**

Der Schaltungssimulator im Altium Designer basiert auf einem tatsächlichen Mixed-Signal-Simulator. Das bedeutet, dass mit diesem Simulator, welcher vom Georgia Tech Research Institute (GTRI) entwickelt wurde, sowohl analoge als auch digitale Geräte analysiert werden können. Der hier besprochene Simulator benutzt eine weiterentwickelte Version des Event-Driven XSPICE, die schon in Abschnitt 2.2.3 vorgestellt wurde. Der XSPICE Simulator im Altium Designer basiert auf dem SPICE3 Berkeley-Code. Er ist voll kompatibel mit SPICE3f5 und unterstützt eine Reihe von Simulationsvarianten sowie XSPICE und PSPICE. Die wesentlichen Funktionen des Simulators im Altium Designer sind folgende: [Alt08a]

### **Feststellung der Simulationsschaltung und Verbinden des Simulationsmodells**

Die integrierte Bibliotheks-Technologie im Altium Designer enthält bereits die schematischen Symbole der entsprechenden Simulationsmodelle. Daher können die schematischen Symbole direkt für eine Simulation der Schaltung verwendet werden.

### **Hinzufügen eines externen Simulationsmodells**

Altium bietet eine große Anzahl von Simulationsmodellen, welcher aber für die Schaltung zur Vervollkommenung des Simulationsmodells Bibliothek noch zusätzliche Modelle hinzugefügt werden müssen. Der Altium Designer Simulator arbeitet einerseits mit einem eigenem Simulationsmodell, andererseits ermöglicht das Simulationswerkzeug aber auch den Import oder

die Integration externer Modelle oder Standards direkt in das eigene Simulationsmodell Bibliothek.

### **SimCode™ für die Simulation digitaler Geräte**

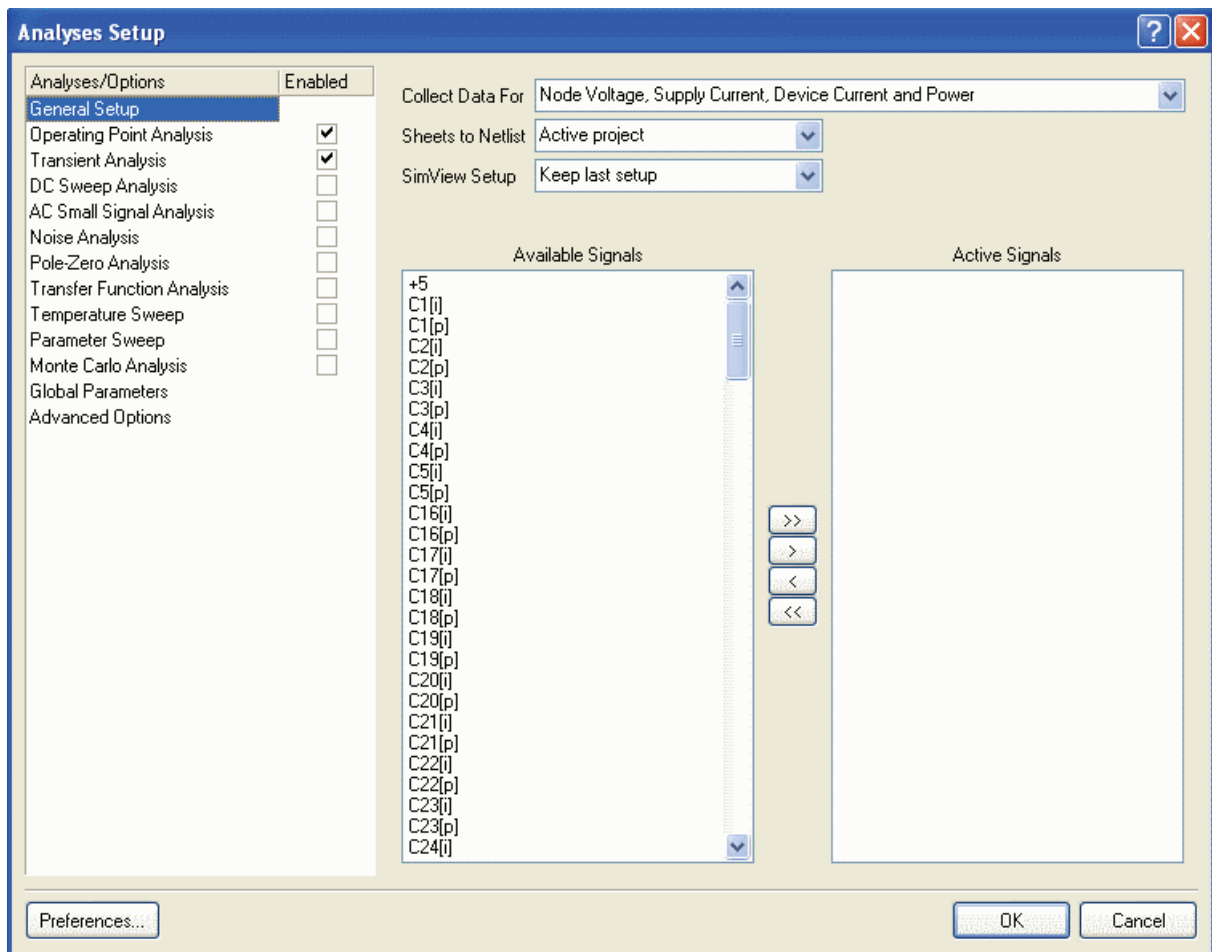
In Altium Designer wurden alle Modelle der digitalen Geräte mit Digital SimCode™ Sprache programmiert. Sie ist eine erweiterte Version von Event-Driven XSPICE und eine Art der C Sprache, die das Verhalten und die Eigenschaften des digitalen Geräts beschreibt. Der SimCode™ wurde aus dem Standard XSPICE Code Modell abgeleitet und in den ASCII Code definiert.

### **Weitere Simulationsanalysen im Altium Designer**

Neben den Simulations-Analysearten, die in Abschnitt 2.3 schon erwähnt wurden, kann der Simulator im Altium Designer noch weitere verschiedene Formen der Signalanalyse durchführen, wie z.B. DC-Sweep-Analyse, Pole-Zero-Analyse, Gleichstrom-Kleinsignal-Analyse, usw.

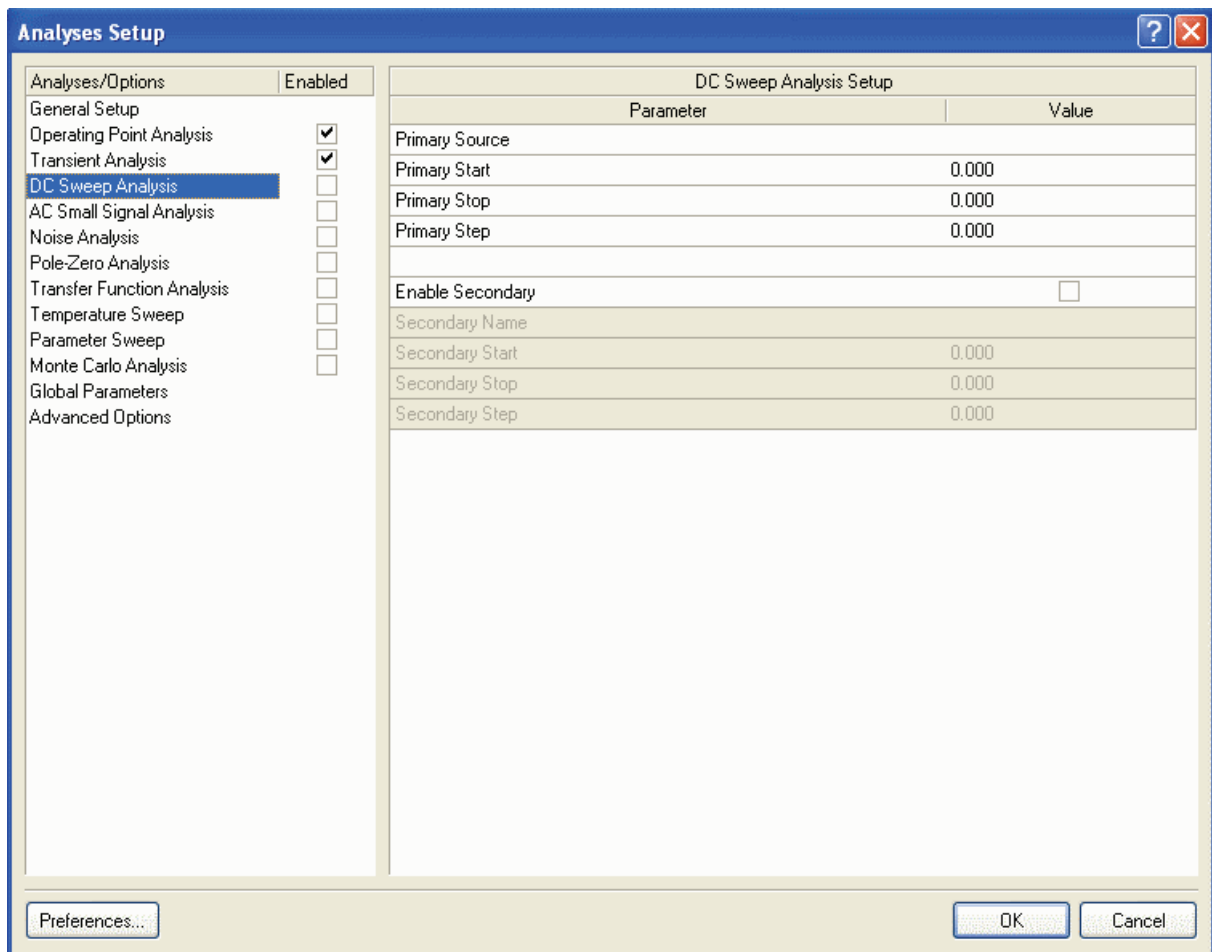
- **DC-Sweep-Analyse** ist ein DC-Charaktere: Durch die Veränderung des Eingangssignals innerhalb eines bestimmten Bereichs wird eine Kurve ausgegeben. Mittels Durchführung einer Reihe von DC-Arbeitspunkt-Analysen wird die gewählte Quell-Signalspannung geändert, und man bekommt eine DC-Transfer-Kurve.
- **Impedanz-Analyse:** Die Impedanz-Analyse beschreibt die Impedanz-Eigenschaften zwischen zwei beliebigen Quell-Terminals. Normalerweise wird sie nur zum Teil von der Kleinsignal-Analyse betrachtet.
- **Gleichstrom-Kleinsignal-Analyse:** Berechnung des DC-Eingangswiderstandes, des Ausgangswiderstandes und des Gain-Wertes für jeden Knoten.
- **Pole-Zero-Analyse** in den einzelnen linearen Input/Output Systemen: Berechnung der Pole oder Nullstelle unter Verwendung der Wechselstrom-Kleinsignal-Übertragungsfunktion, womit die Stabilität der Schaltung analysiert wird.

In der Umgebung des aktuellen Schaltplans kann die Mixed Sim Funktion für die Mixed-Signal-Simulation durch Auswahl des Menüs **Design \ Simulate \ Mixed Sim** aufgerufen werden. Der Simulationsvorgang wird vom Altium Designer automatisch übernommen. Alle Simulationsergebnisse werden jeweils in unterschiedlicher Wellenform aufgezeigt.



**Abbildung 2-2: Simulationsanalysenarten**

Im **Analyses Setup** Fenster, wie in Abbildung 2-2 ersichtlich, können alle Funktionen zur Simulation eingerichtet werden, einschließlich der möglichen Simulationsanalysenarten, des Simulationsumfangs (sheets to Netlist) und der aktiven Signale, welche von der verfügbaren Signalliste ausgewählt und von uns beobachtet werden müssen. Durch Klicken auf den Namen der Simulationsanalysentypen kann der entsprechende Parameter verändert werden (siehe Abbildung 2-3).



**Abbildung 2-3: Parameteränderung für verschiedene Simulationstypen**

Mixed-Signal-Simulation ist eine funktionale Simulation unter Umgebung des Schaltplans. Wenn eine funktionale Simulation für aktuelle Schaltpläne ausgeführt wird, sind dafür alle Komponenten im Schaltplan erforderlich. Jede der entsprechenden Simulationsmodell-Dateien muss vorhanden sein, sonst kann nicht simuliert werden.

Für unseren Entwurf ist die Mixed-Signal-Simulation auf PCB Level leider nicht realisierbar. Hier fehlen einige technische Voraussetzungen, z.B. die Simulationsmodell-Datei aller selbst erstellten Komponenten und der SimCode für die selbst gebauten digitalen Komponenten (PCI Express X4 98 PIN Slot, Crystal Oszillator, usw.).

## 2.6 Board-Level-Simulation im Altium Designer

In den meisten Fällen ist die SPICE Simulation für eine Board-Level-Simulation sehr gut geeignet. Leider hat SPICE auch einige wesentliche Nachteile. So ist z.B. die Simulationsschwindigkeit bei SPICE langsam und nicht alle SPICE-Simulatoren sind voll kompatibel. Um rechtzeitige und genaue Simulationen durchführen zu können, wurde das IBIS-Modell, das alle Komponenten des Systems repräsentiert, als Standard-Modellgerät für die meisten

System-Designer und Halbleiterhersteller entwickelt. Das IBIS-Modell hat eindeutige Vorteile gegenüber SPICE, nicht nur hinsichtlich der kürzeren Simulationslaufzeiten; es ist auch deutlich effizienter (insbesondere bei komplexen Bauteilen wie z.B. Prozessoren). Darum ist das IBIS Modell eine Alternative zur SPICE-Simulation.

### 2.6.1 IBIS

IBIS<sup>12</sup> ist als ANSI/EIA-656 standardisiert und wird auch als ein spezielles Verhaltensmodell in Betracht gezogen. Ein in ASCII-Text-Format formuliertes IBIS Verhaltensmodell enthält I-V<sup>13</sup> und V-T<sup>14</sup> Daten für Input und Output Buffer eines Gerätes. Diese Daten beschreiben die analoge Verhaltenscharakteristik der Ein- und Ausgänge der Geräte (Schaltverhalten, Induktivität, Kapazität, Strom- und Spannungsverhalten), ohne interne Prozesse oder Architekturinformationen des Gerätes zu verraten. Ein IBIS Verhaltensmodell wird manchmal auch als eine "IBIS-Datei" oder ein "IBIS Datenblatt" bezeichnet. Es wird häufig für Signal-Integritäts-Simulationen von High-Speed-Boards oder Systemen benutzt.

Für den schnellen Datenaustausch digitaler Daten zwischen ICs wird ein LVDS Standard<sup>15</sup> entwickelt. Mehr zum Thema LVDS Technik folgt in Kapitel 4 unter Abschnitt 4.1 und Kapitel 8 unter Abschnitt 8.1. Die Anwendungen von LVDS erfordern hohe Datenrate, niedrige Versorgungsspannung, Störungsempfindlichkeit und geringe Störungserzeugung. Die IBIS-Spezifikation wurde fortentwickelt, um diese neuen Standards zu unterstützen. Die IBIS-Spezifikation unterstützt den differentiellen I/O durch differentielles Pin-Mapping. Dadurch erfahren die Simulatoren, die mit IBIS-Dateien anwenden, welche Pins ein differentielles Paar sind und schließlich als invertierenden und nicht-invertierenden Pins beschreibt.

---

<sup>12</sup> Input Output Buffer Information Specification

<sup>13</sup> Strom/Spannung

<sup>14</sup> Spannung/Zeit

<sup>15</sup> LVDS ist standardisiert unter der Bezeichnung TIA/EIA-644. Der zugehörige Verkabelungsstandard ANSI/TIA/EIA-644-1995 mit dem Titel „Electrical Characteristics of Low Voltage Differential Signaling (LVDS) Interface Circuits“ beschreibt die elektrischen Eigenschaften der LVDS Schnittstellen. Weitere Informationen über LVDS stehen in (Kapitel 4 unter Abschnitt 4.1) zur Verfügung.

Das IBIS Dateiformat kann in drei Spalten mit Bezug auf Prozess, Temperatur und deren Schwankungen dargestellt werden. Dazu zählen typische, minimale und maximale Elemente eines IBIS Modells: [IBI02]

- Elemente Pakete (C\_pkg, R\_pkg, L\_pkg) für entsprechend Widerstand, Induktivität und Kapazität.
- (Power\_Clamp) und (GND\_Clamp) beschreiben die Strom-Spannungseigenschaften der Ground- und Spannklemmensschaltung. Wenn die Ausgangsspannung über VCC ist, wird (Power\_Clamp) auf aktiv gesetzt. Entsprechend liegt die Ausgangsspannung unter Ground, wenn (GND\_Clamp) aktiv ist.
- Durch die Lastkapazität C\_COMP wird das zeitliche Verhalten des I/O Modelles begründet.
- In Output-Modellen wird das Treiberverhalten durch Pullup/Pulldown V/I-Tabellen beschrieben. Pullup beschreibt den Pullup Zustand der Buffer, dabei liegt die Ausgangsspannung des Gerätes auf High. Im Gegensatz dazu wird die Ausgangsspannung während Pulldown auf Low gezogen.
- Eine Rampe-Rate beschreibt die Spannungsanstiegsgeschwindigkeit für die Pullup- und Pulldown-Geräte.

Außerdem besteht eine Standard-Modell IBIS-Datei aus drei Abschnitten:

1. Header Info enthält grundlegende Informationen über die IBIS-Datei und welche Daten sie liefert.
2. Komponenten, Pakete und Pin Info enthält alle Informationen über das jeweilige Geräte-Paket, Pin Liste, Pin Betriebszustände und Pin-to-Buffer-Mapping.
3. V-I Behavioral Model enthält alle Daten zur Nachbildung der I-V Kurven sowie V-T Übergangswellenformen.

Dieser Abschnitt soll dem Leser Grundwissen und Grundbegriffe des IBIS Modells vermitteln. Er dient als Ausblick für die späteren Experimente. Anschließend werden in diesem Kapitel einige experimentelle Versuche unter Verwendung des IBIS Modells durchgeführt.



## 2.6.2 BSDL

BSDL<sup>16</sup> ist eine Standard-Modellierungssprache für Boundary-Scan-Test-Fähigkeiten JTAG-kompatible elektronische Bausteine. Der Boundary-Scan-Test ist ein Testverfahren zur Feststellung der defekten Bausteinpins oder defekter Verbindungen auf einer Leiterplatte für elektronische Bausteine. Der erste IEEE 1149.1-1990 Standard zur Beschreibung des Boundary-Scan wurde im Jahr 1990 genehmigt und veröffentlicht. Gegenwärtig BSDL schon ein Teil des 1149.1-2001 „IEEE standard test access port and boundary-scan architecture“.

BSDL Syntax basieren auf einer vereinfachten VHDL-Sprachsyntax, es ist eine Unterart von VHDL zur Beschreibung der Implementierung der Boundary-Scan-Test-Fähigkeiten eines bestimmten elektronischen Bausteins. Die BSDL-Datei wird von Boundary Scan-Tools verwendet, um die Funktionalität des Geräts zu testen. Dabei wird es möglich, z.B. ein Testprogramm oder eine Fehlerdiagnose zu generieren oder eine Prüfbarkeitsanalyse durchzuführen.

*Eine BSDL-Datei enthält eine Auflistung aller im Baustein vorhandenen Boundary Scan-Zellen und beschreibt, wie diese mit den Bausteinpins verbunden sind, welche Funktion sie haben und wie sie in einem Boundary Scan Test eingesetzt werden können. Die BSDL-Datei enthält außerdem die von dem Baustein unterstützten JTAG-Befehle und ihre OpCodes. Oft sind auch noch Angaben über bei der Nutzung des JTAG-Ports zu beachtende Besonderheiten enthalten. Dies können zum Beispiel besondere Reset-Bedingungen sein. [Wik09b]*

In Altium Designer lässt sich eine IBIS oder BSDL-Datei einfach für einen entsprechenden Baustein importieren. Danach kann Altium Designer auf jedes JTAG kompatible Bauteil über die zugehörige IBIS-Datei oder BSDL-Datei zugreifen, die vom Anbieter des Bauteiles bereitgestellt wird, und die Ergebnisse zum Pin Status anzeigen. Das Importieren eines IBIS Modells wird in Kapitel 8 unter Abschnitt 8.2.1 beschrieben.

## 2.6.3 Signalreflexion

Heutzutage nimmt die Anwendung digitaler High-Speed-Geräte immer schneller zu. Solche Geräte arbeiten mit einer schnellen Signal Edge Rate. Schnelle Anstiegszeiten gekoppelt mit einem langen Routing können zu Signalreflexionen führen. Reflexionen auf einer bestimmten Übertragungsstrecke können offenbar nicht nur die echten Signal Daten beeinträchtigen, welche diese Strecke durchlaufen. Sie können auch zum "Rauschen" auf den benachbarten Leitungen – sogenannten elektromagnetischen Interferenzen (EMI) - führen.

---

<sup>16</sup> Boundary Scan Description Language

Wenn Signale entlang der Leitung übertragen werden, kommt es in jedem Moment zu einer Beeinflussung durch eine Transienten-Impedanz. Die Impedanz kommt wahrscheinlich von der Übertragungsleitung selbst. Es könnte auch sein, dass sie aus der Mitte oder vom Ende anderer Komponenten kommt. Wenn die Impedanz konstant ist, dann wird das Signal normalerweise nach vorne verbreitet. Solange die Impedanz sich verändert, werden auch immer - verursacht durch einen auf halbem Wege auftretenden Widerstand, Kapazität, Induktivität, usw. - Signal Reflexionen auftreten. Reflexionen führen zum Über- oder Unterschwingen des Signals.

Ein wichtiger Indikator für die Messung des Anteils der Signals Reflexion ist der Reflexionsgrad. Er wird wie folgt definiert: [Kis03]

$$\rho = \frac{Z_2 - Z_1}{Z_2 + Z_1}$$

Dabei steht  $Z_1$  für die Impedanz vor der Änderung.  $Z_2$  bedeutet die veränderte Impedanz. Dabei wird die Veränderung des Reflexionsgrades in folgenden Zuständen beschrieben:

- Keine Last ( $Z_2 = \infty$ )

In diesem Fall wird der Wert der veränderten Impedanz auf unendlich gesetzt. Dadurch beträgt der Reflexionsgrad = 1. Das bedeutet, dass die rücklaufende Welle gleich groß ist wie hinlaufende Welle.

- Kurzschluss ( $Z_2 = 0$ )

Beim Kurzschluss beträgt die veränderte Impedanz = 0. Der Wert des Reflexionsgrades ist -1. Daraus ergibt sich, dass die rücklaufende Welle negativ zur hinlaufenden Welle ist.

- Korrekter Abschluss ( $Z_2 = Z_1$ )

Der Reflexionsgrad wird auf 0 berechnet. Es gibt keine rücklaufende Welle.

Aufgrund der Existenz des Phänomens der Reflexion ist die Spannung auf dem Punkt, auf welchem die Impedanz-Änderung geschieht, nicht mehr gleich der ursprünglichen Übertragungsspannung. Diese Reflexionsspannung ändert die Wellenform des Signals.

## 2.6.4 Übersprechen

Übersprechen bedeutet: Während der Übertragung des Signals auf die Freileitung entsteht eine ungewünschte Spannung oder Strom Störeinstrahlung über den benachbarten Leitungen durch die elektromagnetische Einkoppelung (z.B. Kopplung durch ohmschen Widerstand, kapazitive Kopplung und induktive Kopplung).

## 2.6.5 Reflexions- und Übersprechenssimulation

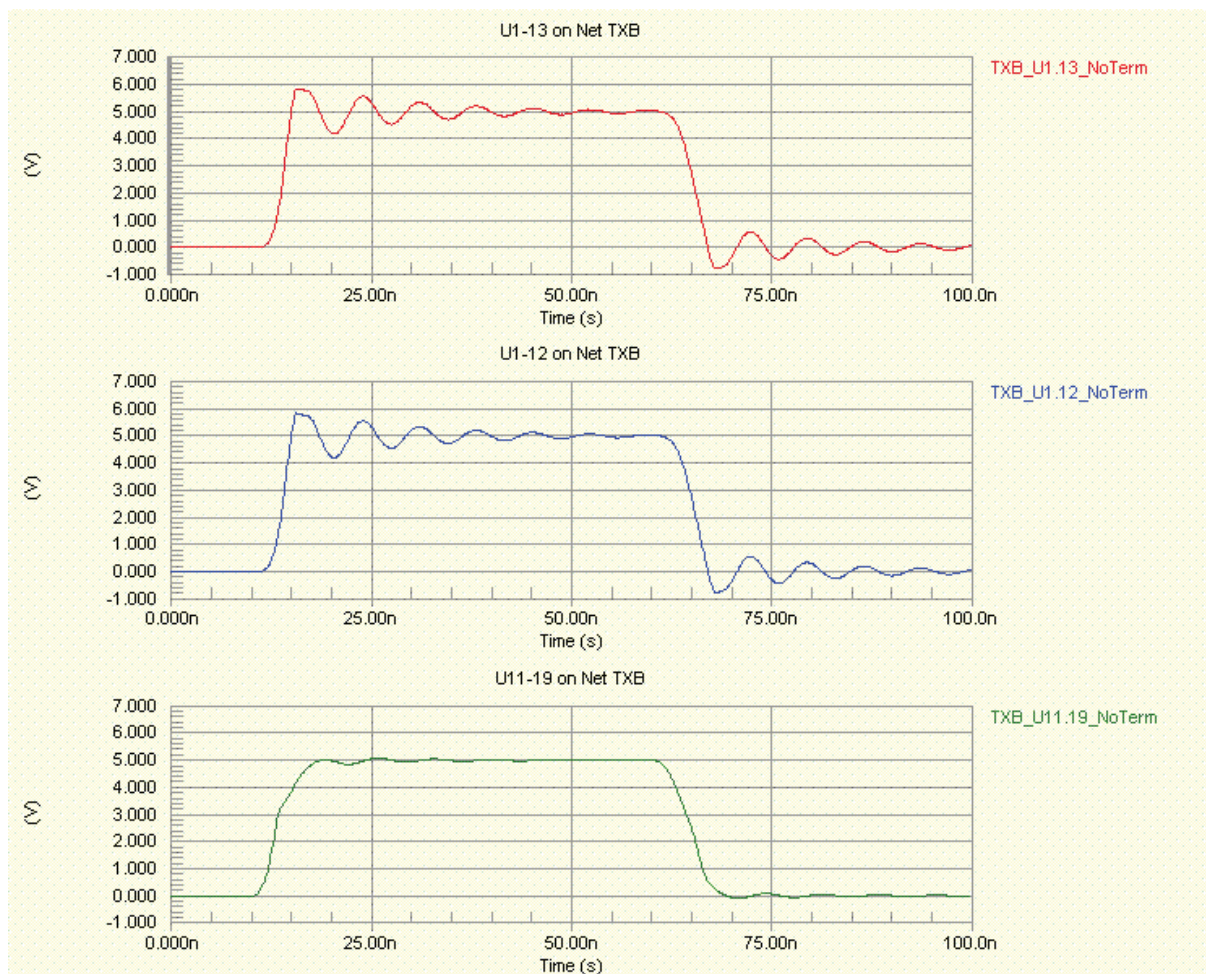
Zur Überwachung und Steuerung der Signalreflexionen und der Cross-Signal-Interferenz (Übersprechen), wird ein Werkzeug benötigt, das die Ausdehnung der Reflexionen und das Übersprechen in Verbindung mit einem Signalentwurf im Detail analysiert. Diese Aufgabe wird in Altium Designer durch die Signal-Integrität-Analyse erfüllt.

Im Altium Designer können die Signal Integrität Analysen sowohl während der Entwurfserfassung als auch der Board-Layout Phasen des Entwurfs durchgeführt werden. Dabei ist es möglich, eventuelle Problembereiche zu identifizieren, die richtige Terminierungsstrategie zu bestimmen und die notwendigen Komponenten für den Schaltplan hinzuzufügen, bevor er auf PCB-Layout gebracht wird. Durch die Vereinigung der Reflexions- und Übersprechenssimulatoren erzeugt der Analysator eine genaue Simulationen für Signalreflexionen und Übersprechen auf dem entsprechenden PCB-Board-Level.

Während beim Reflexionssimulator die Spannungen an den Knotenpunkten eines Netzes mit Routing- und Layer-Informationen berechnet wird, erlaubt es ein Übersprechenssimulator, die Interferenzen zwischen den gekoppelten Netzen zu analysieren.

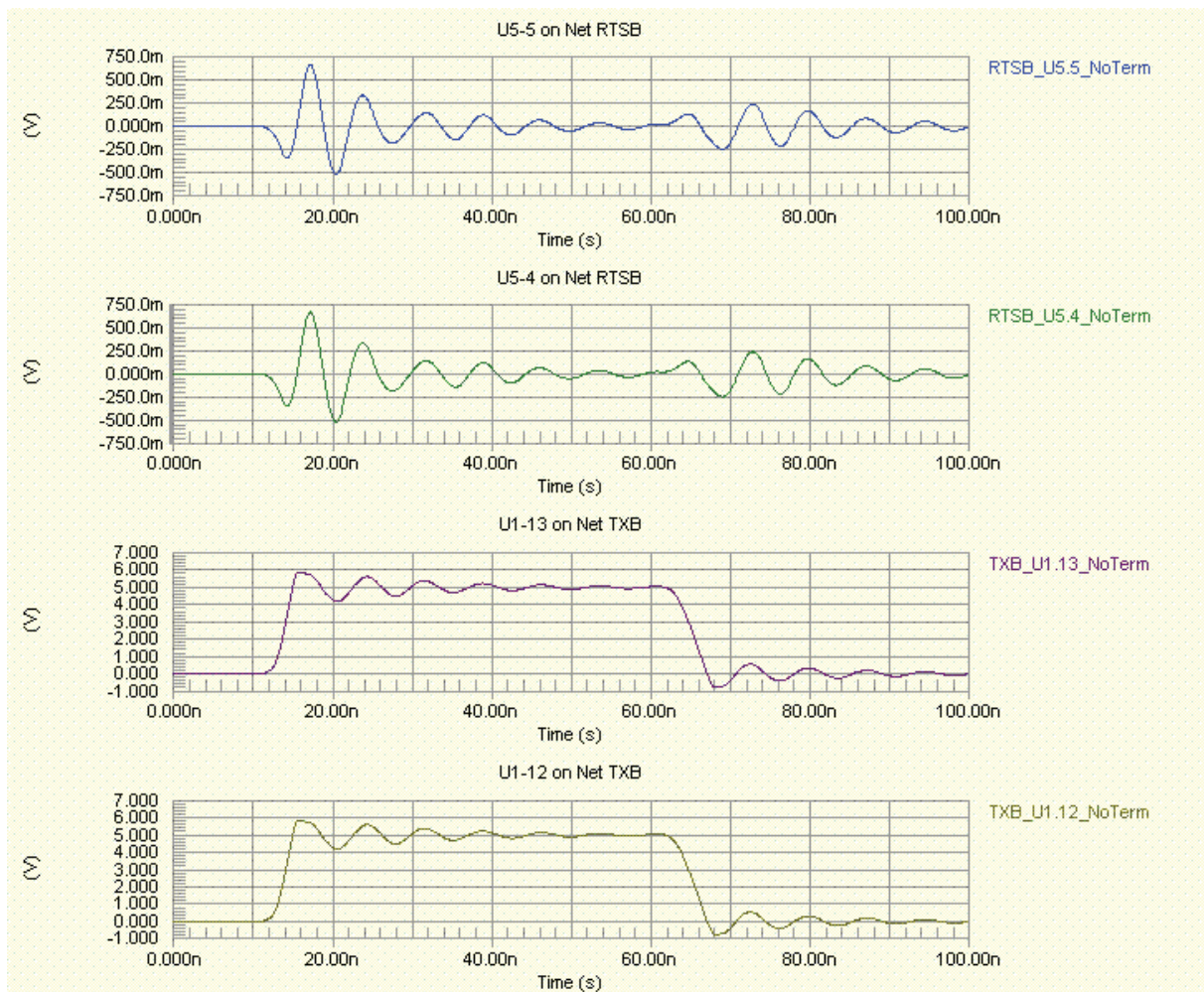
In Abbildung 2-4 wird eine Reflexions-Analyse für das Netz TXB erstellt und dieses mit den folgende Komponenten Pins verbunden: Komponenten U1 Pin 12 und 13, Komponenten U11 Pin 19. Die Signalreflexion führt zu Veränderungen der Impulsform und zum Über- oder Unterschwingen der Spannung auf dem betrachteten Netz. Die Auswirkungen einer Signalreflexion auf die Signalform des betrachteten Netzes TXB hängen nicht nur vom Reflexionsgrad  $\rho$ , sondern auch von der Signalanstiegszeit, der Abfallzeit und von der Leitungslänge (Signallaufzeit) ab. Die Gesamtspannung des Netzes TXB soll während der Signalübertragung zwischen 0 mV und 5000 mV liegen. Die X Achse beschreibt die Signallaufzeit. Übrigens orientiert sich die Y Achse an der elektrischen Spannung. Die Impulsform des mit Komponente U1 Pin 12 verbundenen TXB Netzes lässt sich leicht daran erkennen, dass sich die Gesamtspannung wegen der Signalreflexion durch die reflektierte Spannung um ungefähr

900 mV erhöht oder um 800 mV vermindert. Darum betragen die positiven Überschwingungen des Signals ungefähr 5900 mV, bzw. die Unterschwingungen -800 mV. Mit Hilfe der unterschiedlichen von Altium Designer angebotenen Terminierungsstrategien (z.B. Terminierung mit Reihenwiderstand) können die Überschwingungen und Unterschwingungen deutlich reduziert werden. Darüber hinaus nähert sich die veränderte Signalform der ursprünglichen Form an (Approximation).



**Abbildung 2-4: Reflexionsanalyse (Darstellung aus [Alt08a])**

Die Anzeige der Übersprechenszeichnung ist im Wesentlichen die gleiche wie eine Reflexionszeichnung. Abbildung 2-5 zeigt an Hand eines Beispiels, dass dabei zwei Netze, nämlich TXB (als Angreifer) und RTSB (als Opfer), in einer Analyse des Übersprechens berücksichtigt werden. Das gemessene maximale Übersprechen auf der dem Opfer Netz RTSB angeschlossenen Komponente U5 Pin 5 liegt ungefähr bei 700 mV. Dieses Übersprechen resultiert aus der Reflexion auf das Angreifer Netz TXB während der Signalübertragung, die von Low nach High und von High nach Low wechselt.



**Abbildung 2-5: Analyse des Übersprechens (Darstellung aus [Alt08a])**

Die genauere Implementierung der Signalreflexionsanalyse und Analyse des Übersprechens auf unserem Projekt wird in Kapitel 8 erklärt.

## 2.7 FPGA-Level-Simulation in Altium Designer

Altium Designer wurde vor allem mit einer starken Entwurfseingabefunktion ausgerüstet. Für die FPGA Entwurfseingabe können drei Eingabemethoden verwendet werden: Sprachliche Entwurfseingabe, symbolische Schaltplan-Entwurfseingabe oder Hybrideingabe. Hier wird der hierarchische Entwurfsansatz für den FPGA Entwurf angewendet.

Bei der normalen Softwareprogrammierung kann man sein Programm meistens schnell übersetzen und ausprobieren. Bei einem nicht lauffähigen Programm lässt man der Debugger anlaufen. Dabei wird ein Breakpoint gesetzt und der Inhalt von Variablen, Speichern und Prozessorregistern nachgeprüft, um dem Fehler zu lokalisieren.

Wegen der komplexen Logik des Entwurfs ist ein normaler Debugger für die Untersuchung der FPGA Synthese unrealistisch. Stattdessen wurde ein Logikanalysator verwendet (extern oder als FPGA Soft Core). Doch auch der Einsatz eines Logikanalysators zu Fehlersuche ist sehr mühsam und erfordert viel Geduld.

In Altium Designer wird ein zusätzlicher VHDL-Programm-Testbench (siehe Abschnitt 2.4.2) für jedes Modul entwickelt. Auch sind einige VHDL-Simulatoren mit einer Simulationsfunktion ausgerüstet. Ein echter Testbench lässt sich dadurch aber nicht ersetzen. Damit soll sich die Funktionalität des Modules vor der Synthese mittels Simulation prüfen lassen. Während der Simulation in Altium Designer wird der Testbench auf die obersten Ebene einer Entwurfshierarchie gegeben. Aus Simulatorensicht gibt es keine Unterscheidung zwischen den Teilen des getesteten Entwurfs und dem Testbench selbst.

Die einfachste Testbench ist die Überprüfung einer Abfolge von Schaltungseingaben, deren Operationen in der Schaltung durch die Simulation beobachtet werden können. Ein solcher Testbench besteht aus einer Komponenten-Erklärung und einer Beschreibung der Eingangs-Impulse.

Das folgende Beispiel aus [Alt08a] zeigt die einfachste Form eines Testbenchs. Hier wird die Operation eines NAND-Gatters geprüft. Dabei wird die wichtige Syntax mit Bindestrich kommentiert.

```
library ieee; --Laden ieee 1164 Bibliothek
use ieee.std_logic_1164.all;
use work.nandgate; --Verwenden dieser NAND Gate model von 'work'
--Dieser entity-Beschreibung von der Testbench bleibt Leer
entity testnand is
end testnand;
architecture stimulus of testnand is
--Moduldeklaration
    component nand
        port (A,B: in std_logic;
              Y: out std_logic) ;
    end component;
--Lokal Signalendeklaration
    signal A,B: std_logic;
```

```

    signal Y: std_logic;

begin
--Erstellen einer Instanz für die Komparatorschaltung
    NAND1: nandgate port map (A => A,B => B,Y => Y) ;

--Definieren einer Prozesseslaufzeit
process
    constant PERIOD: time := 40 ns;

begin
    A <= '1' ;
    B <= '1' ;

    wait for PERIOD;

    assert (Y = '0')

        report "Test failed!" severity ERROR;

    A <= '1' ;
    B <= '0' ;

    wait for PERIOD;

    assert (Y = '1')

        report "Test failed!" severity ERROR;

    A <= '0' ;
    B <= '1' ;

    wait for PERIOD;

    assert (Y = '1')

        report "Test failed!" severity ERROR;

    A <= '0' ;
    B <= '0' ;

    wait for PERIOD;

    assert (Y = '1')

        report "Test failed!" severity ERROR;

    wait;

end process;

end stimulus;

```



### 3 Die Grundlage der verwendeten Hardware

Das Kompaktsystem (siehe Abbildung 3-1) wurde von Dipl.-Inf. Falk Berger in Rahmen seiner akademischen Forschungsarbeit Ende 2006 entwickelt. Die Grundlage dieses Kompaktsystems basiert auf einer Multi-DSP<sup>17</sup>-Architektur mit seriellen Hochgeschwindigkeitsverbindungen.

*Dieser Architektur besteht aus einer Basisplatine und je DSP-Modul einer Steckkarte. Die DSP-Module befinden sich auf den Steckplatinen, welche senkrecht in PCI-Slots der Basisplatine stehen. Diese Anordnung der Module ermöglicht kurze Leiterbahnen zwischen den Modulen und ist daher vorteilhaft für die Kommunikationswege. [Hir07]*



**Abbildung 3-1: Überblick auf dem Kompaktsystem**

#### 3.1 Das DSP-Modul D.Module.6713 von D.SignT

D. Module.C6713 [Dsi04] wurde mit dem eine erweiterte VLIW<sup>18</sup> Architektur besitzenden Gleitkomma-fähigen Digitalen Signal Prozessor TMS320C6713 von Texas Instruments bestückt. Dieser moderner 32 Bit Signalprozessor arbeitet in Gegensatz zum Typ der Vorgängermodule (- D.Module.C6701) mit einer erhöhten maximalen Taktrate bis zu 300 MHz. Dieser RISC DSP bietet 8 Befehlseinheiten, die parallel betrieben werden und liefert eine maximale Leistung von 2400 MIPS<sup>19</sup>, 1800 MFLOPS<sup>20</sup>. Dabei wurde auch eine verbesserter

---

<sup>17</sup> Ein Digitaler Signalprozessor kann als hochspezialisierte Ausprägung eines eingebetteten Systems betrachtet werden.

<sup>18</sup> Very long Instruction Word

<sup>19</sup> Million Instructions Per Second

<sup>20</sup> Millions of Floating Point Operations per Second



DMA<sup>21</sup> Controller und ein 16 Bit breites Host Interface in diesen Prozessor eingebaut. In Abbildung 3-2 sieht man die Komponenten, die sich auf dem DSP-Modul D.Module.6713 befinden.

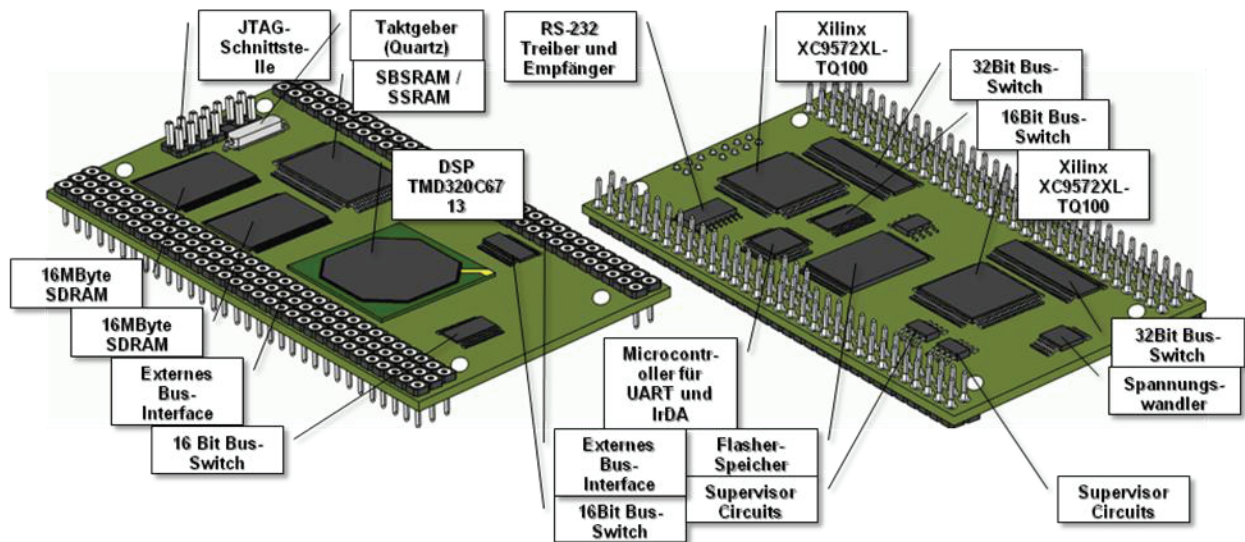


Abbildung 3-2: Überblick zum DSP-Modul D.Module.6713

### 3.1.1 Die Speicher

Der DSP selbst bietet bis zu 256 KByte internen Speicher, geeignet für die Anwendung der Programm oder Daten. 64 KByte dieses Speicherbereichs kann konfiguriert werden als 4 Wege Second Level Cache um den Datendurchsatz zu erhöhen, wenn sich große Datenstrukturen oder Programm Code im externen Speicher befinden.

Die 512 KByte synchroner SRAM<sup>22</sup> (SBSRAM<sup>23</sup>) arbeiten mit 100 MHz Taktfrequenz und bieten einen Einzel-Zyklus-Zugriff.

Zusätzlich können 32 MB On Board SDRAM<sup>24</sup> kann für große Daten Puffer verwendet werden, wie sie häufig für Bildverarbeitungsanwendungen erforderlich sind. Er wird verwendet

<sup>21</sup> Direct Memory Access

<sup>22</sup> Static Random Access Memory

<sup>23</sup> Synchronous Burst Static Random Access Memory

<sup>24</sup> Synchronous Dynamic Random Access Memory

für Programmcode und Daten und arbeitet mit dem EMIF<sup>25</sup> Takt. SDRAM arbeitet auch bei 100 MHz.

Zwei MByte Flash Speicher sind vorgesehen, um nicht flüchtige Daten und Programme zu speichern. Die Flash Speicher ist in mehrere Sektoren von 64 Kbyte geteilt. Jeder Sektor kann auf Byte Ebene individuell gelöscht und neu programmiert werden.

### **3.1.2 Die externe Busschnittstelle**

Das D.Module.C6713 verfügt über eine externe Bus-Schnittstelle [Mic08], die für Speicher und I/O Erweiterungen zuständig ist. Dazu zählt:

#### **Schnittstellen des C6713 (On-Chip)**

Das 16 Bit breite TMS320C613 Host Port Interface (HPI) ist voll erreichbar auf dem D.Module. Ein Host Kontroller hat direkten Zugriff auf den gesamten Speicher bereich des DSP.

Das External Memory Interface (EMIF) ist der Bus für den Speicherzugriff, der 32 Daten- und 20 Adressleitungen besitzt.

Das Multichannel Buffered Serial Port (McBSP<sup>26</sup>) ist eine bidirektionale serielle Schnittstelle mit DMA-Kanälen, wovon jeder DSP zwei McBSPs besitzt.

#### **DSP-spezifische Signale**

4 externe Interrupt-Signale

Takt-Signale (CPU, EMIF, ...)

32 Bit Timer: Ein Oscillator auf dem Board erzeugt den DSP Mastertakt. Weitere Takte für externe Peripheriegeräte können generiert werden bei Benutzung der beiden 32 Bit Timer.

---

<sup>25</sup> Beim „External Memory Interface“ (EMIF) handelt es sich um einen speziellen Speicherbus integrierter Schaltkreise (z.B. Mikroprozessoren), über den sich zusätzlicher externer Speicher ansprechen und somit in die Kernanwendung einbeziehen lässt.

<sup>26</sup> „Multichannel Buffered Serial Port“ ist ein proprietärer Schnittstellenstandard der DSP-Familie C6000 von Texas Instruments.

## Modulspezifische Schnittstellen und Signale

Standard RS232-Port (COM-Port)

Inter Integrated Circuit-Bus (I<sup>2</sup>C)

RESET und Watchdog: Ein externer Reset Input kann benutzt werden um ein System-reset-Signal anzuschließen, oder eine Taste für den manuellen Reset. Eine Watchdog bietet Sicherheit gegen Programmabstürze und Hardware ausfällen.

Der programmierbare CPLD<sup>27</sup>-Logikbaustein besitzt 72 programmierbare Makrozellen und ist speziell für Peripherie bzw. für das Einbinden des Modules in Hostsysteme vorgesehen.

16 freie E/A<sup>28</sup>-Signale "General Purpose I/O" (GPIO) können als Input, Output, oder bidirektionale Signale konfiguriert werden.

### 3.2 Das Basisboard

Das Basisboard verfügt über eine externe Busschnittstelle, darauf kann das DSP Module eingesteckt werden. Außerdem wurde es mit einem programmierbaren Logikbaustein (FPGA<sup>29</sup> Cyclone II EP2C8) der Firma Altera ausgerüstet, und es befindet sich in der Mitte des Basisboards. Die Cyclone II FPGA-Familie der Firma Altera besteht aus Bausteinen mit bis zu 68614 Logikelementen und bis zu 1.1 Mbits an eingebetteten Speicher. Der hier verwendete FPGA EP2C8 ist der zweitkleinste FPGA dieser Familie. Er hat 8256 Logikelemente. Seine insgesamt 239616 Bits RAM sind in 52 RAM-Blöcken zu je 4 kbits plus 512 Parity-Bits organisiert. Der verwendete FPGA besitzt neben den Logikelementen außerdem noch 18 eingebettete Multiplizierer, stellt dabei LVDS Leitungen (siehe Kapitel 4 unter Abschnitt 4.1 und Kapitel 8 unter Abschnitt 8.1) zur Verfügung. Die LVDS Leitungen wurden durch den Basisboard Steckkartenverbinder<sup>30</sup> realisiert. Abbildung 3-3 zeigt die wichtigen Komponenten auf dem Basisboard.

---

<sup>27</sup> Complex Programmable Logic Device

<sup>28</sup> Eingabe/Ausgabe

<sup>29</sup> Field Programmable Gate Array

<sup>30</sup> Hier wird der PCI Express 4X Steckverbinder mit 98 Pins auf dem Basisboard verwendet.

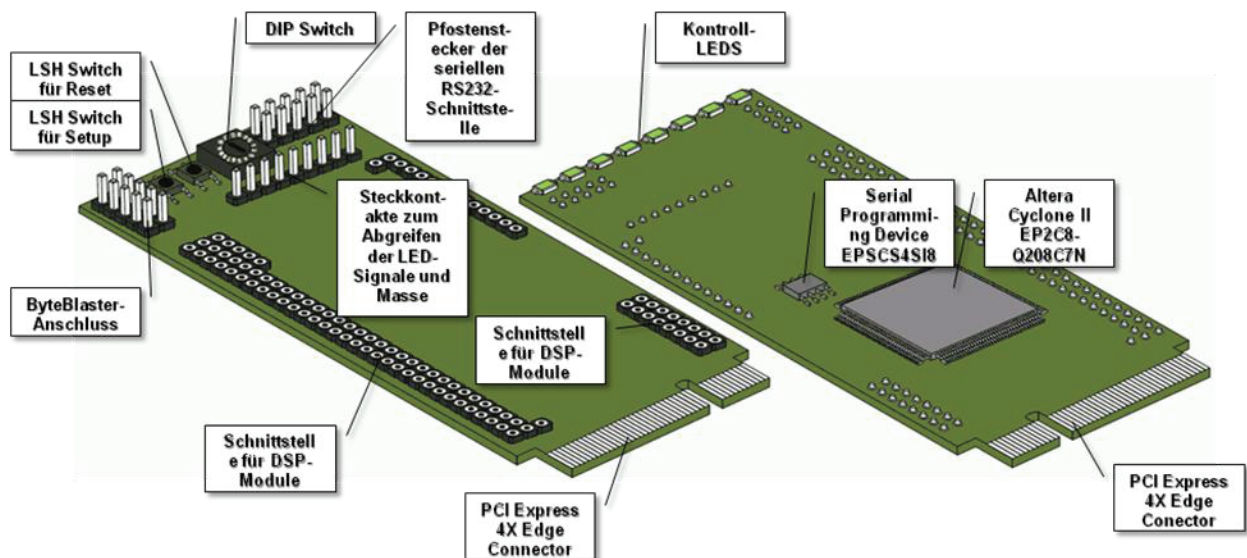
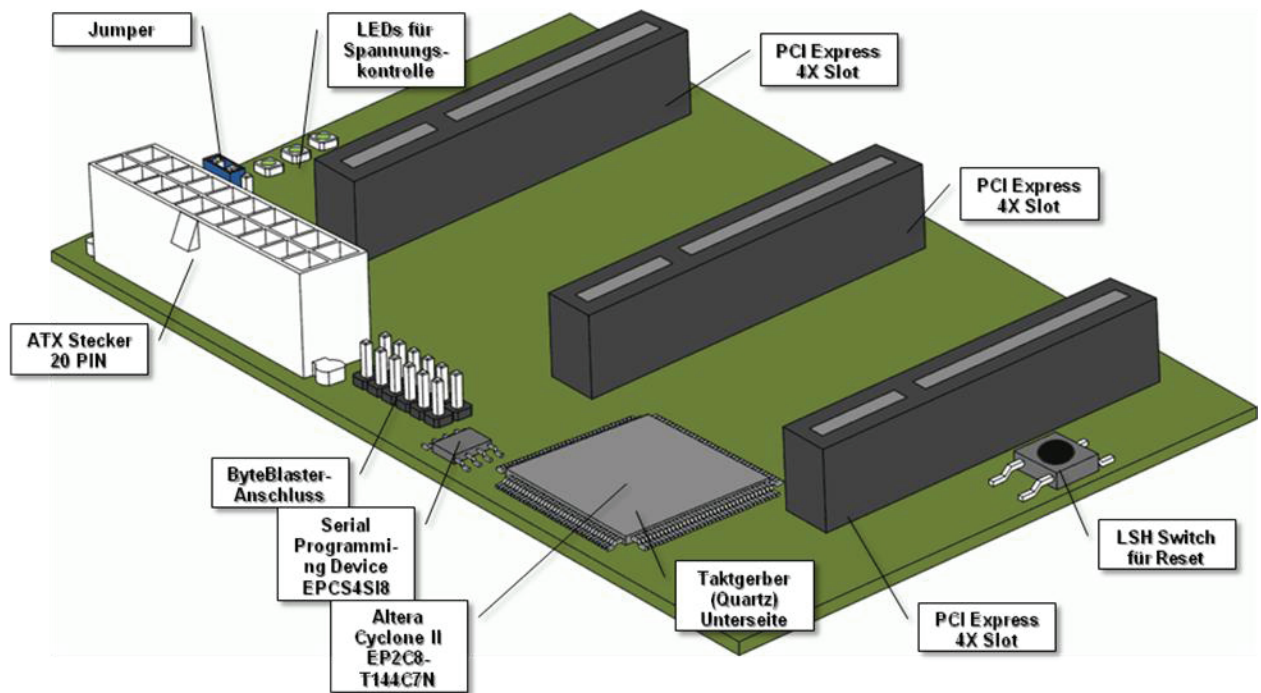


Abbildung 3-3: Überblick zum Basisboard

### 3.3 Die Backplane

Die Backplane trägt drei Steckverbinder<sup>31</sup> für die DSP Basisboards und versorgt diese dabei mit Strom. Die Backplane und ihr programmierbarer Logikbaustein (FPGA) implementieren die drei Kommunikationswege. Mehr zu diesen drei Kommunikationswegen folgt im nächsten Kapitel. Die wesentlichen Komponenten auf der Backplane werden in Abbildung 3-4 vorgestellt.

<sup>31</sup> Auf der Backplane wird der PCI Express 4X Slot mit 98 Pins verwendet.



**Abbildung 3-4: Überblick zum Backplane**

Bei dem ringförmigen Kommunikationsteilsystem sind, die zwei benachbarten DSP jeweils über diese LVDS-Leitungen miteinander verbunden. Die Backplane schließt den LVDS-Kommunikationsring der aufgesteckten Verbundmodule. Neben diesem ringförmigen Kommunikationsweg existiert auf der Backplane noch ein Kommunikationsbus für das McBSP-basierte Kommunikationssystem und den boardseitigen Switch für das McBSP-basierte Kommunikationssystem.

## 4 Kommunikationssysteme des seriellen Multi-DSP-Systems

Der von mir überarbeitete Prototyp verfügt über drei verschiedene DSP zu DSP Kommunikationswege. In der Diplomarbeit [Mül08] von Michael Müller wurden alle drei Kommunikationswege und deren physikalische Strukturen ausführlich beschrieben. Die Backplane wurde nicht weiterentwickelt und entspricht immer noch der ersten Version, in der mit 3 DSP-Modulen gearbeitet wird.

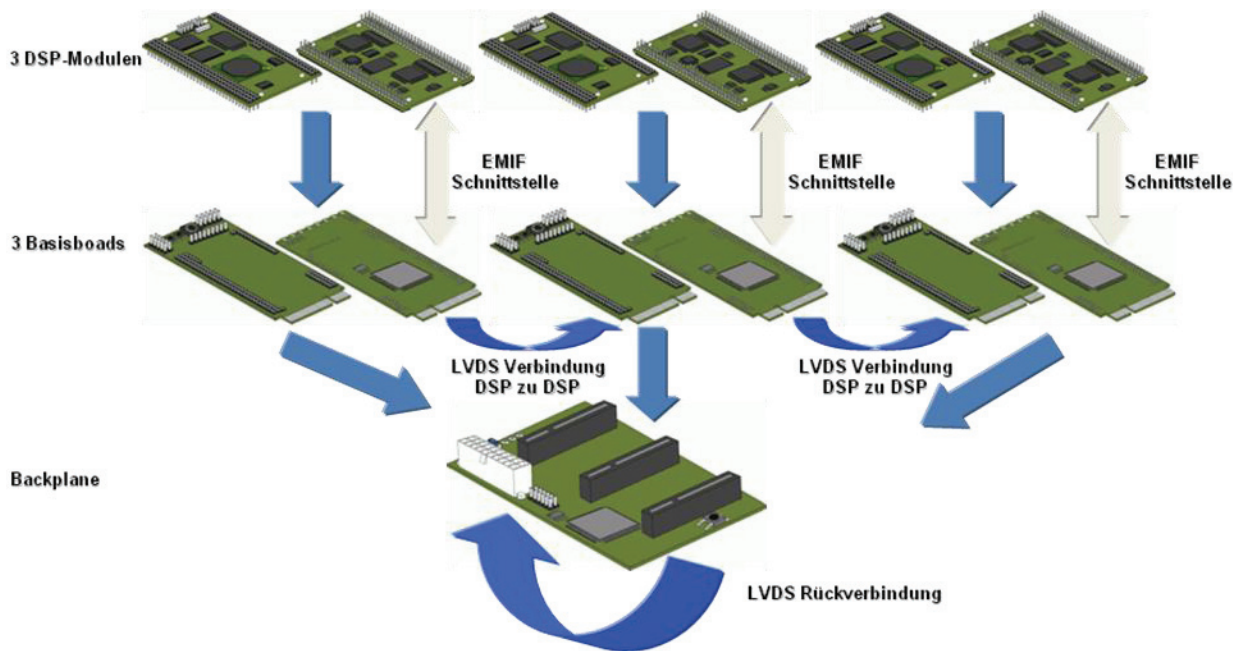
Drei Kommunikationswege lauten:

- McBSP-basiertes Kommunikationssystem mit Switch
- LVDS-basiertes ringförmiges Kommunikationssystem
- McBSP-basiertes Kommunikationssystem mit Bus

Beim McBSP-basierten Kommunikationssystem mit Switch handelt es sich um die Verwendung des Ports 1 der DSP eigenen McBSP Schnittstelle. Der boardseitige Switch der Backplane dient dem Nachrichtenaustausch zwischen zwei DSPs.

Beim McBSP-basierten Kommunikationssystem mit Bus handelt es sich wieder um die Verwendung des Ports der DSP eigenen McBSP Schnittstelle. Aber diesmal betrifft es den Port 0. Jedes DSP Modul ist über diesen Port an einen gemeinsamen Kommunikationsbus angeschlossen.

Bei dem zweiten Kommunikationssystem handelt es sich um einen Punkt zu Punkt Verbindung zwischen jedem DSP und einem eigenen FPGA, wie in Abbildung 4-1 dargestellt. Diese erfolgt über die DSP-seitig verfügbare EMIF Schnittstelle. Der FPGA hat spezielle Ein und Ausgabekanäle und baut eine LVDS Verbindung zwischen zwei benachbarten Basisboards auf. Dabei wurden alle DSPs über diese LVDS Leitungen miteinander verbunden. Eine entsprechende Rückverbindung vom letzten zum ersten FPGA sorgt für einen Ringschluss. Jeder Prozessor muss mithilfe des jeweiligen FPGA in den Ring hinein kommunizieren.



**Abbildung 4-1: LVDS-basiertes ringförmiges Kommunikationsweg**

*Der entscheidende Vorteil dieses Kommunikationssystems liegt in seiner räumlich flexiblen Ausbaubarkeit<sup>32</sup> und den hohen Übertragungsraten<sup>33</sup>, welche sich damit realisieren lassen. [Mül08]*

#### 4.1 LVDS Technik

„Low Voltage Differential Signaling“ in Kurzform LVDS ist ein elektrischer Schnittstellenstandard, mit dem serielle Hochgeschwindigkeitsdatenübertragungen realisiert werden können. Diese Übertragungstechnik kommt bei vielen Halbleiterherstellern zum Einsatz.

Christian Suffa hat in seiner Diplomarbeit [Suf08] beschrieben, dass die LVDS Signale mithilfe differentieller Spannungspegel, welche durch Umpolung der Leitungen erzeugt werden, übertragen werden. Je Übertragungsrichtung und pro Informationsbit wird ein Leitungspaar benötigt.

<sup>32</sup> Mithilfe der LVDS Technik lassen sich große Entfernungen im Bereich von mehreren Metern problemlos überwinden.

<sup>33</sup> Die Datentransfers innerhalb dieses Systems durchlaufen sowohl die EMIF- als auch die LVDS Verbindungen. Dabei muss man die LVDS Technik als begrenzenden Faktor ansehen, da das EMIF als Speicherschnittstelle deutlich höhere Übertragungsraten als auch einen besseren Datendurchsatz bietet. Realistisch sind Übertragungsraten von über 100 MByte/s für das gesamte Kommunikationssystem.

Die LVDS Technik hat eine sehr geringe Leistungsaufnahme und einen niedrigen Stromverbrauch, weil dabei eine Arbeitsspannung mit einem Absolutwert von 1,2 V und einem sehr geringe Signalstrom von 3,5 mA verwendet werden.

Für sehr hohe Übertragungsraten stellt LVDS allerdings auch strenge Anforderungen, da zur Wahrung der Datensynchronität alle zugehörigen Leitungen die gleiche Leitungslänge aufweisen müssen. Dies gilt für alle Übertragungsleitungen.

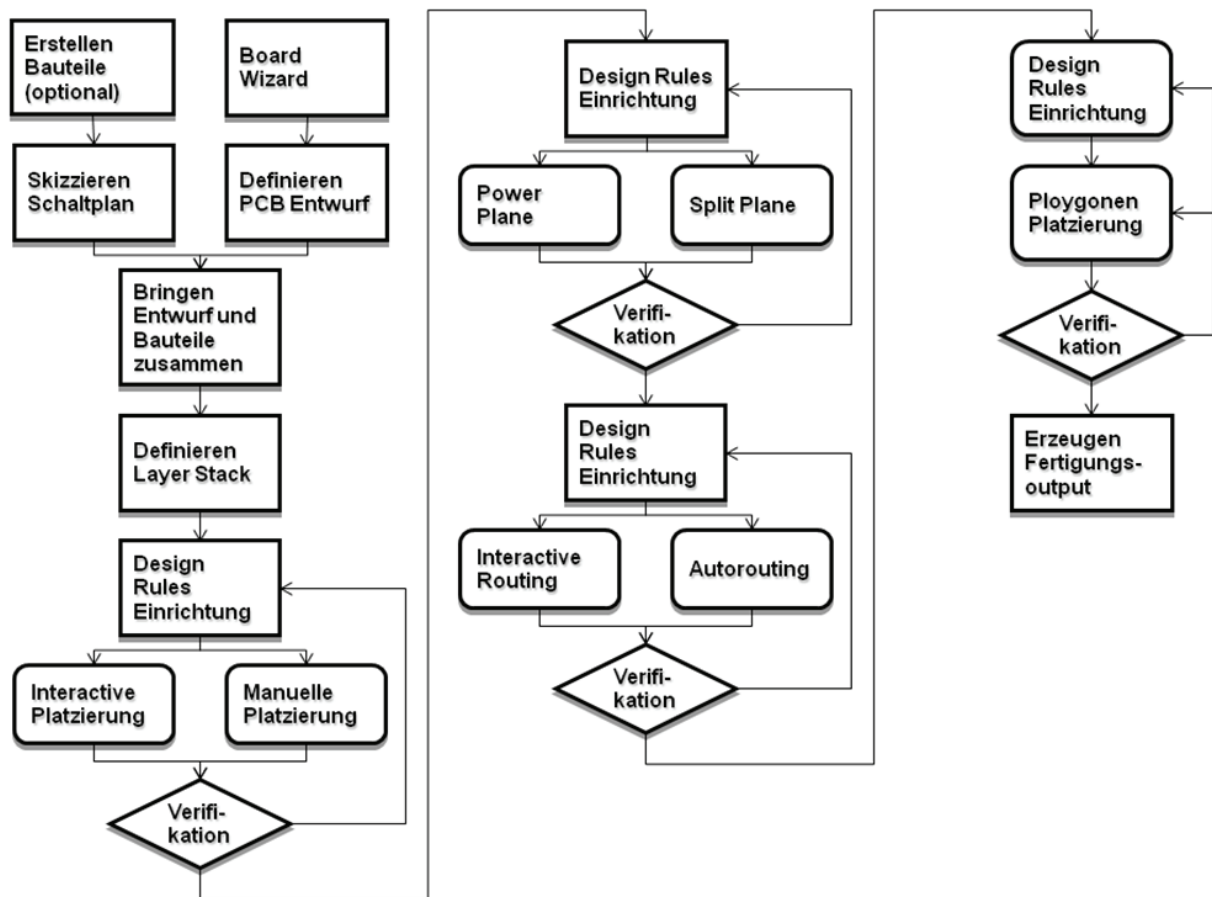
Aus diesem Grund wird im Kapitel 8 unter Abschnitt 8.4 noch genauer vorgestellt, wie man mit Altium Designer die Leitungen so auslegt dass sie die gleiche Länge haben.

Dieser Abschnitt soll einen Überblick sowie Grundwissen zur LVDS Technologie und den entsprechenden Standards vermitteln. Die praktische Verwendung dieser Technik wird in Kapitel 8 unter Abschnitt 8.1 detailliert behandelt.



## 5 Altium Designer

Altium Designer ist der Nachfolger der erfolgreichen Leiterplatten Layout Software Protel. Er ist branchenweit das erste und einzige Programm, welches eine komplette Lösung für Board Level Design bietet. Altium Designer wurde konzipiert, um den gesamten Prozess der Entwicklung von Elektronikprodukten zu vereinheitlichen und um die Disziplinen Board Design, Design von programmierbarer Hardware und Softwareentwicklung, die in der Vergangenheit immer getrennt waren, zusammen zu führen. Altium Designer erweitert die traditionellen Grenzen des Board Designs. Layout Routing mit Altium Designer ist komplett regelgesteuert und verwendet dabei Situs topologische Autorouting Funktion, die keine geometrischen Einschränkungen hat. Die Integration von FPGA-Design Funktionen ermöglicht es der System Design Entwicklung, das FPGA- und PCB-Design zu integrieren. Eines der Highlights von Altium ist die Software-to-Hardware-Compilierung. Dabei unterstützt sie FPGA und Board Design und ermöglicht zudem eine gleichzeitige Eingabe von Schaltplänen und der VHDL Hardware Beschreibungssprache im Eingabemodus. Wie bereits im vorherigen Kapitel schon erwähnt, unterstützt Altium Designer neben der VHDL-basierten FPGA Simulation auch die gemixte Signal Schaltkreis Simulation und Layout vor und nachher Signal Integrität Analysis. Zu den Vorteilen dieser Elektronik Designsoftware gehören noch eine gemeinsame Benutzeroberfläche und ein einheitliches Datenmodell. In der folgenden Abbildung 5-1 kann man den Ablauf einer Platinen-Herstellung sehen.



**Abbildung 5-1: Ablauf für eine PCB Layout**

Die folgenden Ausführungen zum Altium Designer basieren auf meiner Studienjahrarbeit [Dai09]. Darin wird die Herstellung der fehlenden Komponenten und Leiterplatten sowie die Platzierung aller Komponenten auf Platinen behandelt. Jedoch werden die beiden Leiterplatten nicht geroutet. Darum dient diese Arbeit auch als Fortsetzung meiner Studienjahrarbeit. Die Schritte von „Design Rules Einstellung“ bis zur letzten Leiterplattenprozedur sind Gegenstand der weiteren Darlegungen.

## 6 Vorbereitung der Leiterplatten Verdrahtung

Vor dem richtigen Leiterplatten Routing Prozess sind noch einige Arbeiten und die Lösung der verbliebenen Probleme nötig. Dieses Kapitel soll einen fundamentalen Überblick zu den im Hintergrund stehenden Vorbereitungsarbeiten zur Verdrahtung der Leiterplatte vermitteln. Zu den sogenannten Vorbereitungsarbeiten gehören eine gründliche Entwurfsregeleinstellung, der Aufbau eines Top-Level-Schemas und die Platzierung der notwendigen Layer und Vias. An dieser Stelle wird darauf hingewiesen, wie wichtig die elektrische Regelprüfung und die Entwurfsegelprüfung für die Leiterplattenproduktion sind.

Einige durch die Entwurfsregelprüfung entstehende Probleme werden im Mittelteil dieses Kapitels behandelt. Die entsprechenden Lösungen werden anschließend in Detail beschrieben.

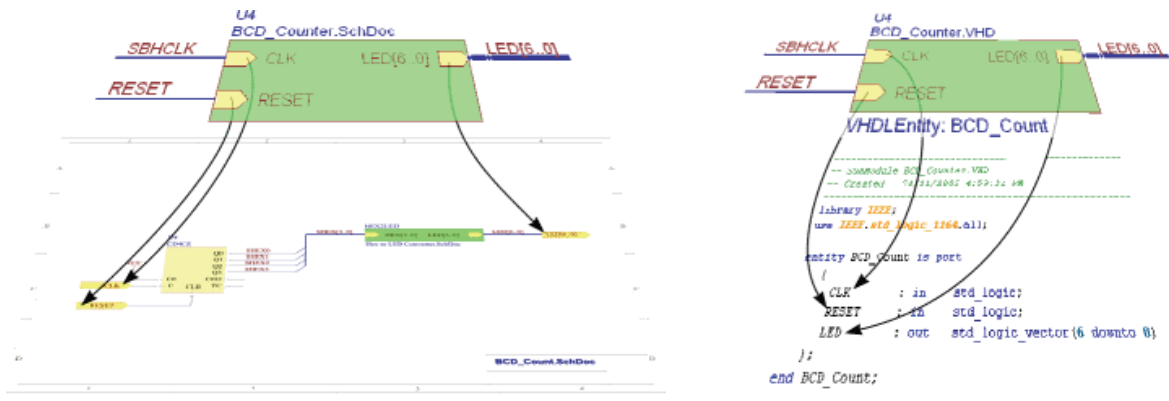
### 6.1 Top-Level-Schematic

Top-Level-Schematic verwirklicht einen hierarchischen Entwurf. Dabei ist es möglich, die Schaltpläne anderer verschiedener Projekt-Dokumente per Verweis in einem Projekt miteinander zu verknüpfen. Der gesamte Entwurf ist in logische Blöcke aufgeteilt, wobei jede Sub-Schicht auf der Top-Level-Ebene durch ein Block-Symbol vertreten ist. Die Beziehung eines Projekt-Dokumentes zum anderen Projekt-Dokument und zu den Netzverbindungen werden durch Informationen im Quell-Projekt definiert.

Die Dateinamen-Attribute auf jedem Symbol der Sub-Schicht verweisen auf die zugrundeliegende Entwurfsdatei, die sie vertreten. Wie in Abbildung 6-1 kann diese Datei dargestellt durch: [Alt08a]

- Eine Schaltplan Schicht
- Ein OpenBus System Dokument
- Eine VHDL oder ein Verilog Datei

Eine schematische Sub-Schicht kann ein Symbol umfassen, das auf niedrigere Entwurfs-Dateien verweist. Mit diesem Ansatz kann eine tiefere und komplexere Entwurfshierarchie erstellt werden.



**Abbildung 6-1: Top-Level-Schematic (Darstellung aus [Alt08a])**

Für unser Projekt wurde ein Top-Level-Schematic für das gesamte Kompaktsystem zur Repräsentation des internen Systemverhältnisses aufgebaut. Diese Top-Level-Schematic besteht aus vier logischen Blöcken. Sie vermittelt genau die drei Basisboards, die sich senkrecht in PCI-Slots auf der Backplane befinden (siehe Kapitel 3 unter Abschnitt 3.2). Die Verbindungen zwischen den logischen Blöcken werden von sechs Bus Systemen erfasst. Sie beschreiben Input oder Output der entsprechenden Logischen Blöcke. Außerdem sind sie auch Teile der LVDS Ring Verbindungen (siehe Kapitel 4 unter Abschnitt 4.1 und auch Kapitel 8 unter Abschnitt 8.1). Übrigens wurden ein paar Änderungen für die Schaltpläne von Basisboard und Backplane vorgenommen.

Erstens sollten für jeden Eingang und Ausgang der logischen Blöcke in der Top-level-Schematic die zugehörigen Input und Output Ports in den Schaltplänen des Basisboard oder der Backplane erstellt werden.

Zweitens besetzen sowohl der PCI Steckverbinder auf dem Basisboard als auch der Steckplatz auf der Backplane je 98 Pins, die auf zwei Seiten aufgeteilt sind. Jeder an den Pins angeschlossenen Leitung wird extra ein neuer Name vergeben. Die neue benannten Leitungen werden später mit dem passenden Bussystem verbunden, das zu den Input oder Output Ports führt. Die Neubenennung der Leitungen verursacht einige Probleme:

- Nach Aktualisierung heißt in Leitplatten Layout die ursprünglichen GND und VCC Leitungen auf der Backplane jetzt A1\_49 und A1\_3. Hingegen heißen die GND und VCC Leitungen auf dem Basisboard A\_49, A\_3.
- Die originale Netzklasse für LVDS Verbindungen muss auch neue definiert werden.

Die beiden Probleme haben allerdings keinen großen Einfluss auf die PCB-Endkonstruktion. Die erstellte Top-Level-Schematic für die Beschreibung des Kompaktsystems befindet sich auf der Begleit-DVD, welche sich nur mit Altium Designer öffnen lässt. Außerdem befindet sich dort eine exportierte grafische Darstellung der Top-Level-Schematic im PDF Datenformat im Verzeichnis **Anhang \ Kompaktsystem**, Im Anhang wurde das Dokument im Extra-Großformat ausgedruckt.

## 6.2 Design Rules (DR), Entwurfsregeln Einstellung

Derzeit werden Platinen relativ schnell zu niedrigen Kosten hergestellt. Diese Herstellung hat einige Einschränkungen, so werden zum Beispiel der Platz und der Abstand auf dem Platinen Layout festgelegt. Diese Einschränkungen können sehr leicht in **PCB<sup>34</sup> Rules and Constraints Editor** Dialog geändert werden. Dieses Kapitel beschreibt die Festlegung der Einschränkungen und deren Einstellungen für unseren Entwurf.

Entwurfsregeln können in folgende Kategorien eingeordnet werden, die im Folgenden erläutert werden.

- Mindestabstand zwischen allen Leitungen, Pads<sup>35</sup> und Komponenten für verschiedene Layer<sup>36</sup>
- Leitungsbreite für verschiedene Verdrahtung
- Routing Via<sup>37</sup> Style
- Matched Netz Length Rules

### 6.2.1 Mindestabstand zwischen allen Leitungen, Pads, Komponenten

Im Layout<sup>38</sup> werden später geometrische Merkmale überprüft, die auf Grund elektrischer Randbedingungen eingehalten werden müssen. Mehr zu Entwurfsregelprüfung folgt in Abschnitt 6.3. Zu den geometrischen Merkmalen gehören beispielsweise erforderliche Abstands-

---

<sup>34</sup> Printed Circuit Board

<sup>35</sup> Lotflächen

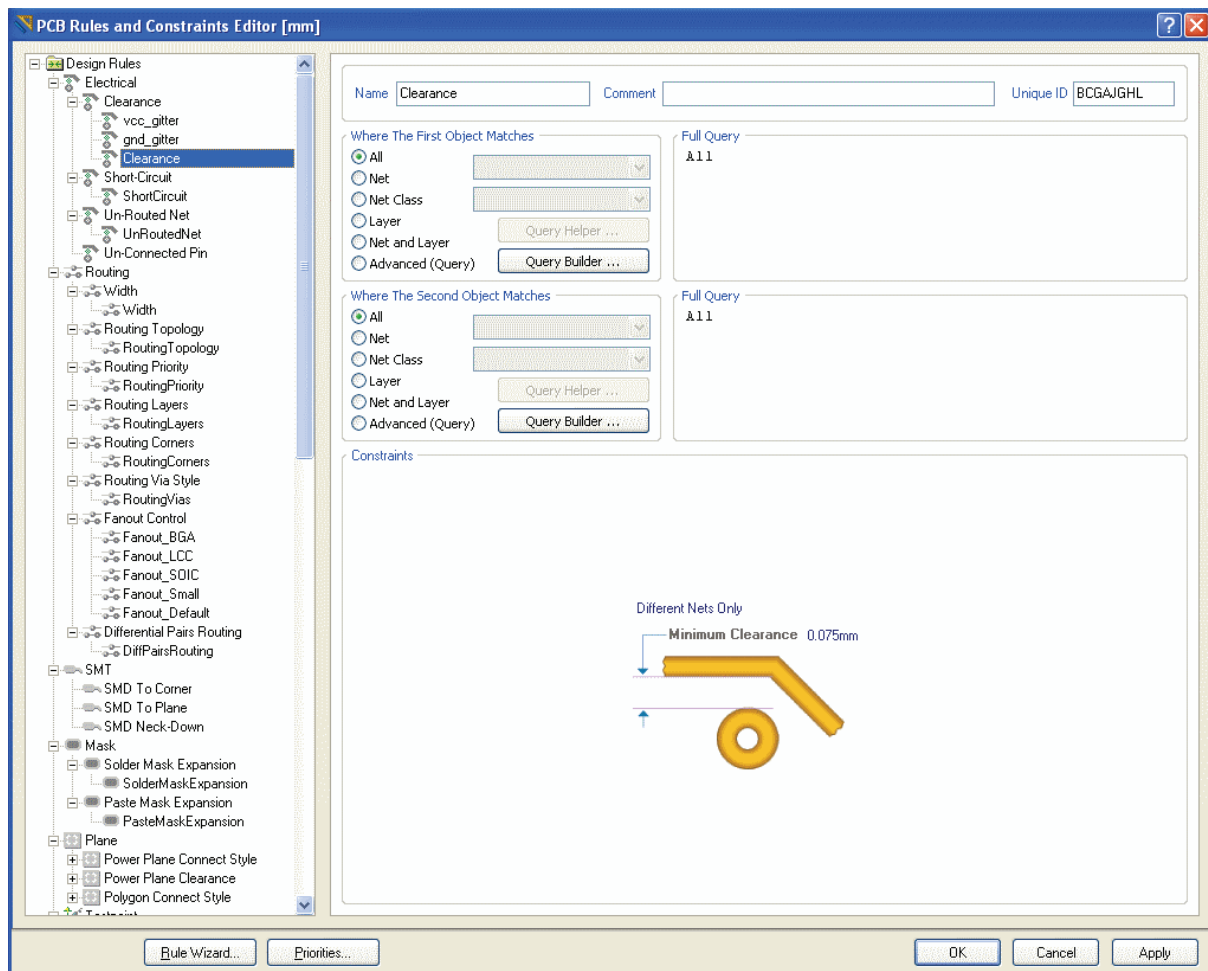
<sup>36</sup> Leitplatte Schicht, unsere Entwurf besitzt vier Schicht (Top Layer, GND Layer, VCC Layer, Bottom Layer). Weitere Beschreibung findet man in Abschnitt 6.6.

<sup>37</sup> Durchkontaktierung

<sup>38</sup> Leiterbahnen Struktur

bedingungen zwischen Bauelementen und/oder Leiterbahnen, um Feldstärken und Kapazitäten in Grenzen zu halten.

Im PCB Editor Fenster wählen **Design \ Rules**, dann Klicken auf **Clearance** unter **Electrical** Kategorie. Hier lässt sich der Mindestabstand ändern, wie in Abbildung 6-2 dargestellt.



**Abbildung 6-2: Festlegung für Mindestabstand**

In diesen Entwurf, sind viele Komponenten sehr dicht auf dem Top- und Bottom Layer platziert. Darum wurde der Mindestabstand von 0,254 mm auf 0,075 mm reduziert. Zusätzlich müssen für die GND<sup>39</sup>- und VCC<sup>40</sup> Layer zwei neue Regeln (gnd\_gitter<sup>41</sup>, vcc\_gitter<sup>42</sup>) einge-

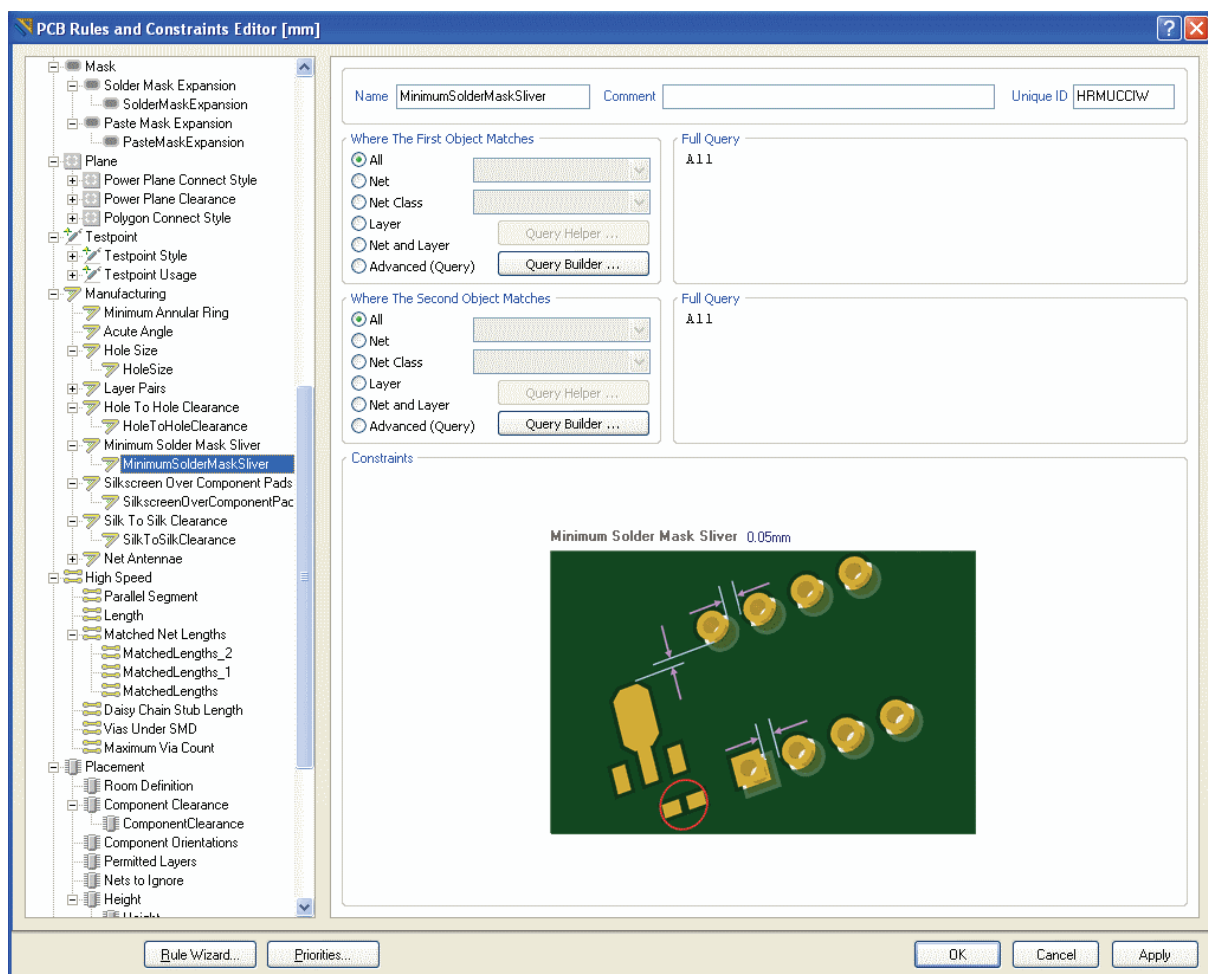
<sup>39</sup> Als GND bezeichnet man einen leitenden Körper, der im Regelfall mit dem Potential 0 Volt definiert wird und das Bezugspotential für alle Signal- und Betriebsspannungen darstellt.

<sup>40</sup> VCC ist der Anschluss für die Versorgungsspannung (Pluspol) einer Schaltung. Wird dieses Symbol mehrfach in einem Schaltplan genutzt, so muss man sich vorstellen, dass alle VCC-Kontaktpunkte miteinander verbunden sind. Durch mehrfache Verwendung erspart man sich das Zeichnen der direkten Leiterbahnverbindungen, was auch wegen der Übersicht oft günstiger ist. Es gibt auch Schaltungen, die mehrere unterschiedliche Spannungen benötigen. Daher gibt es statt VCC auch Symbole wie +UB, +5V, +12V, +VSS usw.

<sup>41</sup> Für GND Layer wurde ein Gitter gebaut um alle GND Kontaktpunkt zu erfassen.

fügt werden. Auf dem GND- und VCC Layer gibt es hingegen viel Platz und der Abstand spielt keine große Rolle. Aus diesem Grund kann ein Abstand von 0,3 mm eingestellt werden.

**Minimum solder mask sliver** unter **Manufacturing** Kategorie hilft dabei, die engen Abschnitte der Lötmaske für eine spätere Herstellung zu identifizieren. Berühren oder Überschneiden sich zwei Objekte der Lötmaske, so wird eine Regelverletzung festgestellt. Dazu gehören alle Pads, Komponenten Pads und Vias (Durchkontaktierung), die sich auf einem gemeinsamen Layer befinden. Ein Wert von 0,05 mm als Minimum solder mask sliver ist für diesen Entwurf bestens geeignet (siehe folgende Abbildung 6-3).



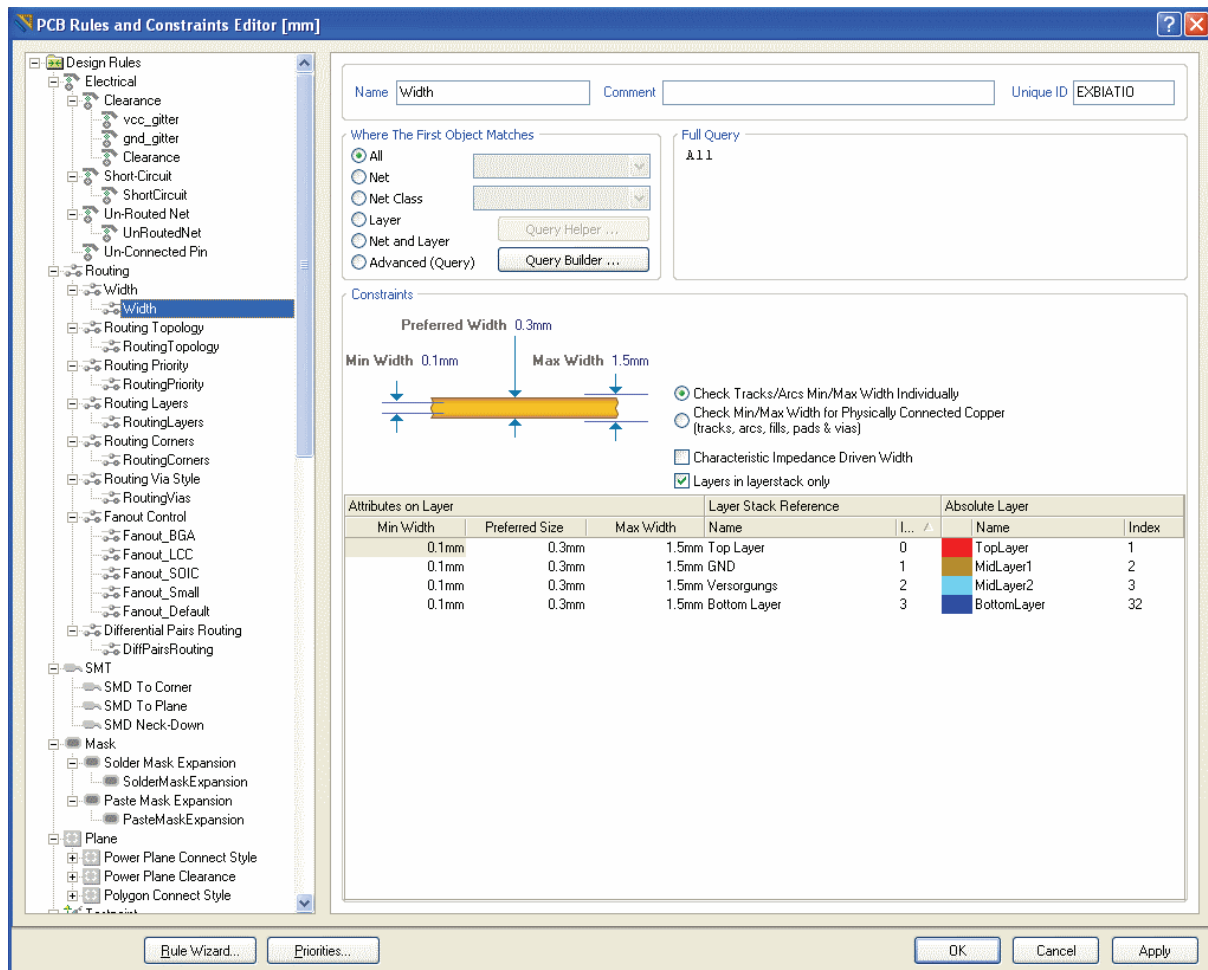
**Abbildung 6-3: Festlegung Minimum solder mask sliver**

## 6.2.2 Leitungsbreite für verschiedene Verdrahtungen

Die Auslegung der Mindestleiterbahnbreite für eine bestimmte Strombelastung, die aus einer höchstzulässigen Stromdichte resultiert, wird in diesem Abschnitt beschrieben. In Abbildung

<sup>42</sup> Für VCC Layer wurde ebenfalls ein Gitter gebaut um alle GND Kontaktpunkt miteinander zu verbinden.

6-4 sieht man, die Veränderung der Leitungsbreite mit Hilfe der Auswahl **Routing \ Width** realisierbar ist.



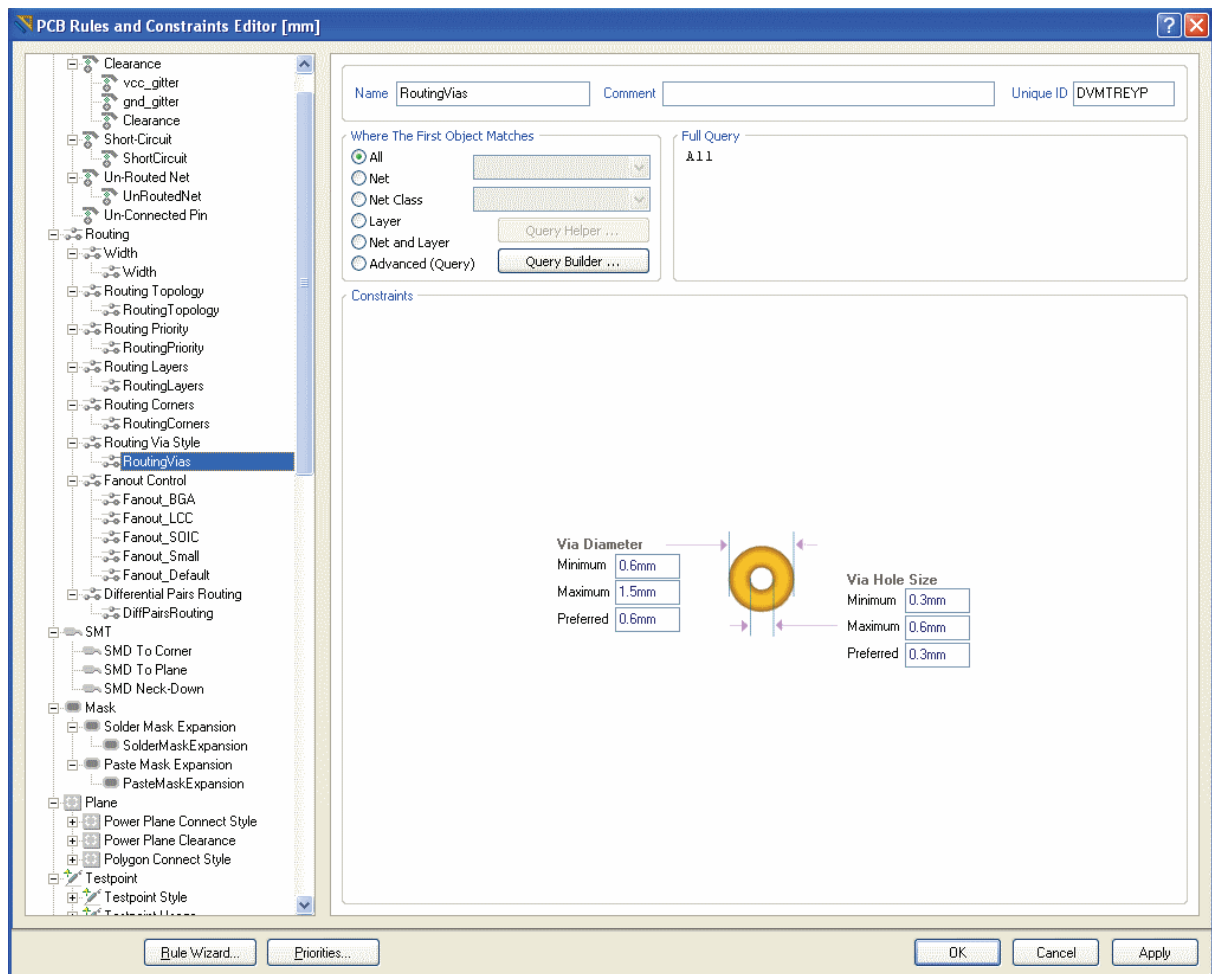
**Abbildung 6-4: Festlegung für Leitungsbereite**

Die dickste in der Arbeit verwendete Leitung hat eine Breite von 1,01 mm. Im Gegensatz dazu ist die dünnste Leitung 0,15 mm breit. Aus diesem Grund wurden die Breiten von 0,1 mm (Minimum Leitungsbreite) bis 1,5 mm (Maximum bereit) gewählt, und die bevorzugte Breite auf 0,3 mm gesetzt. Das gilt für alle Layer. Während des Routing Prozesses kann per TAB Taste das **Interactive Routing for Netz** Dialogfeld aktiviert werden um die Leitungsbreite jeder Zeit ändern zu können, genaue Beschreibung dazu findet man in Kapitel 7 unter Abschnitt 7.2.1.

### 6.2.3 Routing Via Style

Routing Via Style gibt dem Vias die Durchmesser und Lochgrößen an, wie in Abbildung 6-5 vorgestellt.





**Abbildung 6-5: Festlegung Via Style**

In diesem Entwurf wurden die Maximalen, Minimalen und bevorzugten Via Durchmesser auf 1,5 mm, 0,6 mm und 0,6 mm festgelegt. Für die Maximalen, Minimalen und bevorzugten Via Lochgrößen wurde die Werte auf 0,6 mm, 0,3 mm, 0,3 mm festgelegt. Die Werte lassen sich während der Via auf der Platine aufgebracht wird jederzeit mit der TAB Taste ändern. In Abschnitt 6.7 erklärt wie ein Vias auf die Platine platziert wurde.

#### 6.2.4 Matched Net Lengths Rules

**Matched Net Lengths** unter **High Speed Rules** arbeitet mit einer zulässigen Differenz der Leiterlängen. Dabei werden die zu betrachtenden Leitungen mit einander verglichen, wobei die längste Leitung die zu erzielende Länge der restlichen Leitungen vorgibt. Die genaue Einstellung wird in Kapitel 7 unter Abschnitt 7.3.2 erklärt.

### 6.3 Design Rule Checking (DRC), Entwurfsregelprüfung

Im Rahmen der Schaltungsentwicklung können bei der Erzeugung des Layouts Fehler auftreten, die eine volle Funktionsfähigkeit der Schaltung beeinträchtigen. Dazu kommt, dass jeder

Leiterplattenhersteller unterschiedlichen Restriktionen mit Gültigkeit für den jeweiligen Technologieprozess unterliegt.

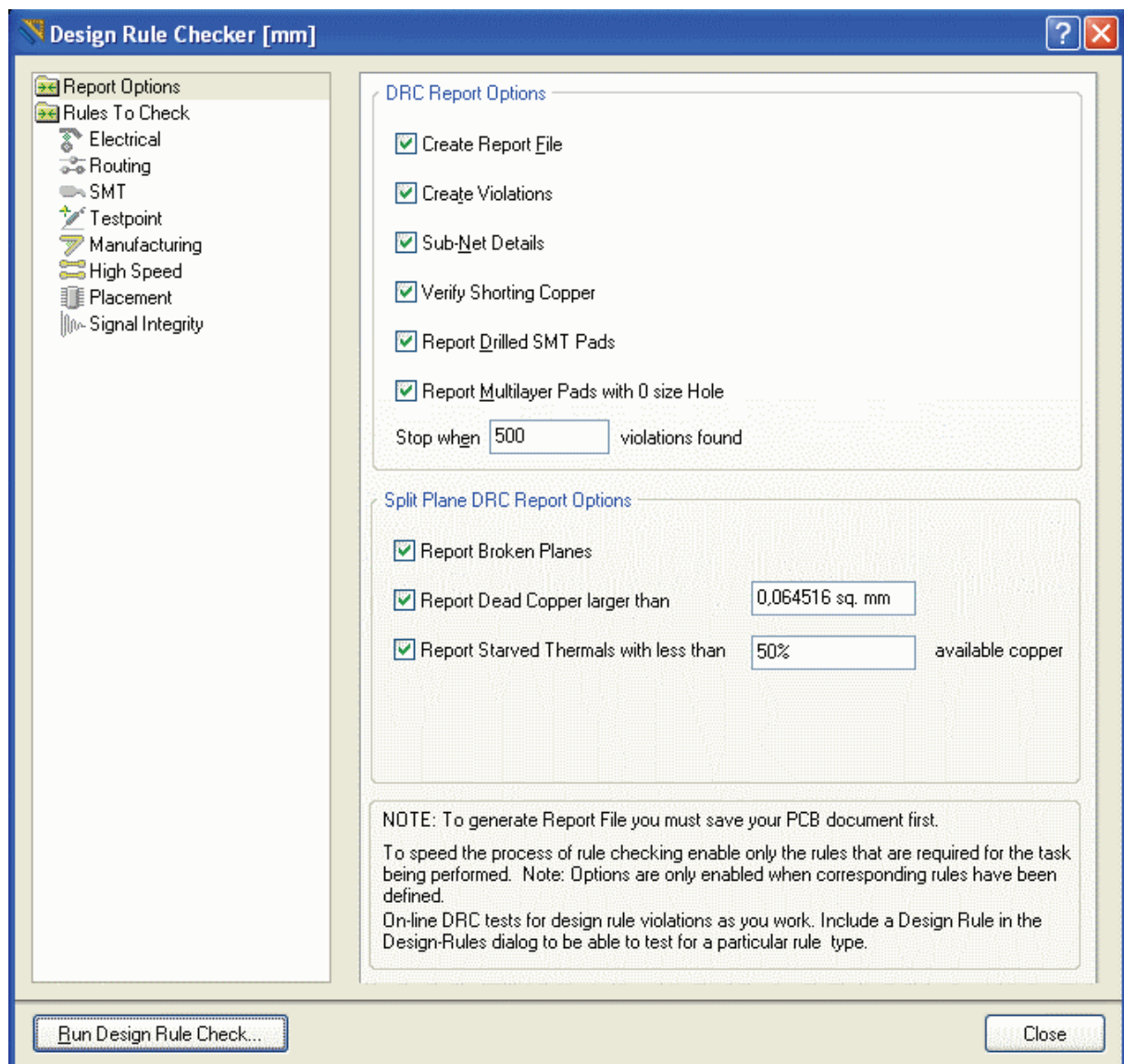
Solche Einschränkungen und Anforderungen an das Layout dienen dem zuverlässigen Funktionieren der hergestellten Schaltung und müssen dem Schaltungsentwickler vom Leiterplattenhersteller vorgegeben werden. Treten Fehler auf, so muss ein Redesign erfolgen, was trotz eines enormen Zeitaufwandes immer noch kostengünstiger ist als die Herstellung einer funktionsunfähigen Schaltung.

Die Überprüfung und Kontrolle des Schaltungslayouts auf Übereinstimmung mit den Vorgaben und Regeln des Leiterplattenherstellers erfolgt automatisch durch entsprechende Programme im Altium Designer.

Bei der Herstellung unterliegen die Leiterplattenhersteller auch einer ganzen Anzahl geometrischer Restriktionen. Die Entwurfsregeln (Design Rules) bestimmen z.B. die Mindestabstände, Mindestbreiten, Überlappungen und Innenlagen der Layer und Layerkombinationen. Neben dem Layout sind die Design Rules Eingangsparameter für die Entwurfsregelprüfung. Regelverletzungen sowie eine Fehlerliste werden im Message Panel in Form eines generierten Berichtes zusammengefasst.

Beim Design Rule Checking (DRC) im Altium Designer handelt es sich um ein leistungsstarkes automatisiertes Tool mit der Aufgabe, die logische wie auch die physische Integrität des Designs zu prüfen und zu überwachen. Im Rahmen dieser Kontrolle sind eine oder alle Design-Regeln aktiviert. Die Überprüfung kann auch online erfolgen.

Design Rule Checking (DRC) Befehl kann aus dem Tools Menü aufgerufen werden, die Konfiguration für die Durchführung in der Design Rule Checking Dialog sieht man in Abbildung 6-6.



**Abbildung 6-6: Design Rule Checking**

In der Ordner Liste auf der linken Seite des Dialogs steht der Design Rule Kategorien, deren Regel Typen können geprüft werden. Abbildung 6-7 veranschaulicht, nach Klicken auf die **Rules to check**, alle überprüfbar Design Rule Typen, in allen Kategorien werden auf der rechten Seite des Dialogs gelistet.

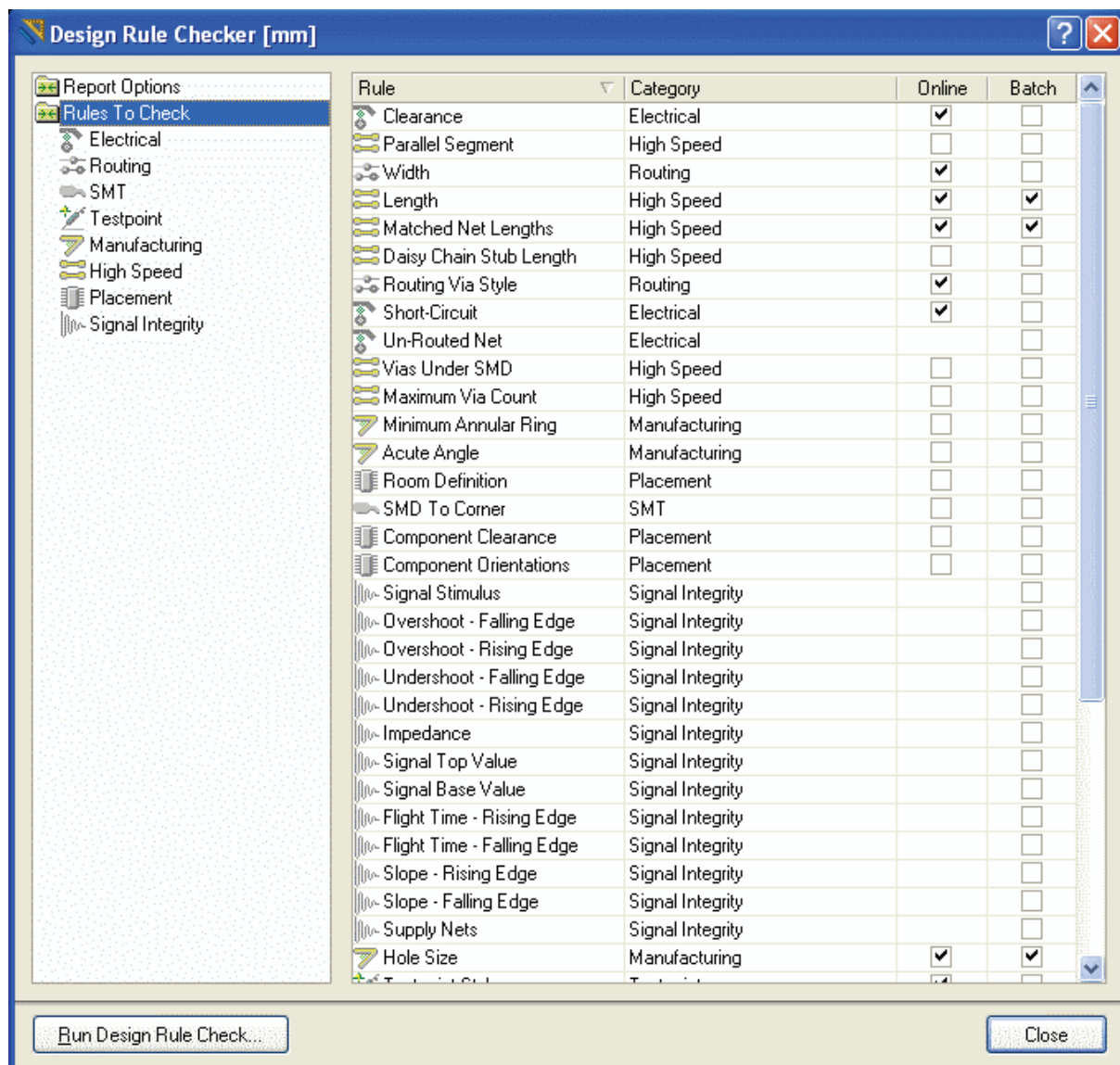


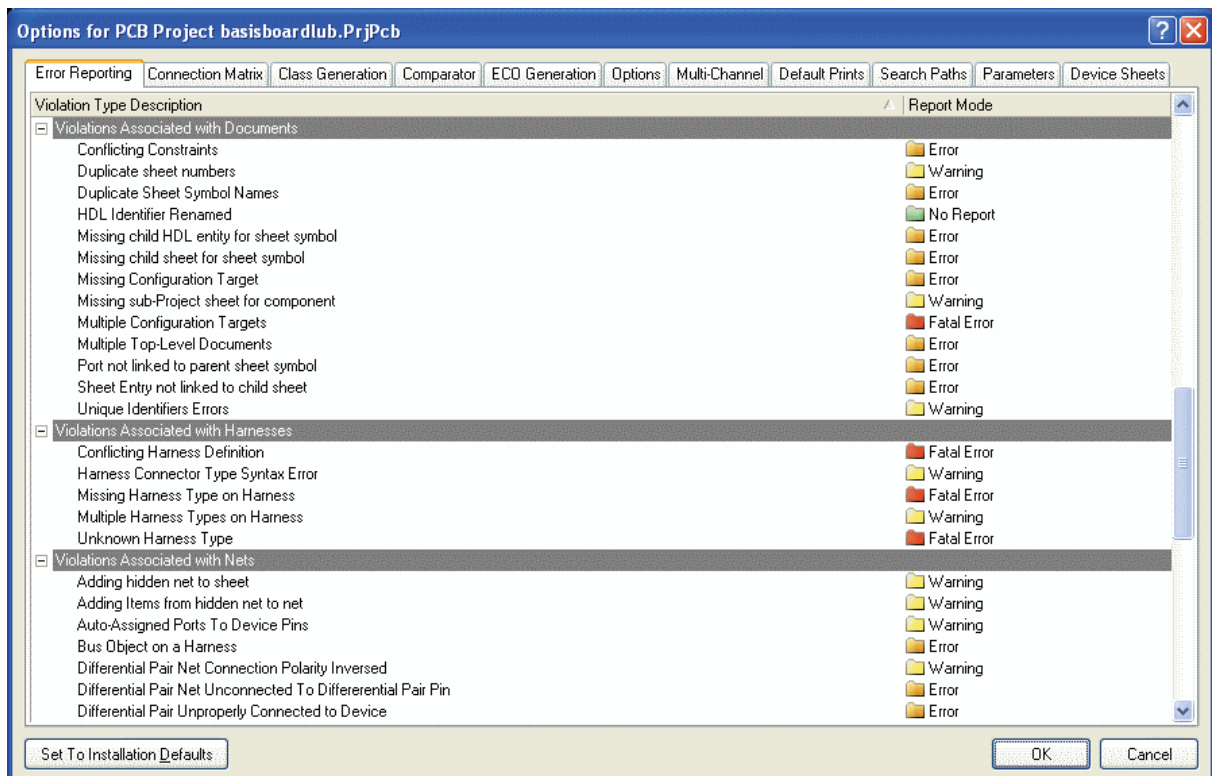
Abbildung 6-7: Design Rule Typen

## 6.4 Electrical Rule Checking (ERC), Elektrische Regelprüfung

Vernetzungsprobleme im Schaltplan sind immer schwer zu finden. Davon sind einige der häufigsten und schwierigsten: Elektrische Fehler, die Verdrahtung der falschen PINs oder einem Netz wurde eine falscher Netzname vergeben. Zur Aufdeckung solcher Fehler dient das elektrische Regeln Checking (ERC). Während dieses Prozesses werden möglicherweise einige elektrische Probleme gefunden und angezeigt, die von Designer bei der Schaltplanherstellung vernachlässigt wurden. Neben typischen Verdrahtungsfehlern werden z.B. auch doppelte Bezeichnungen, gezeichnete Netz Laber, die einander nicht berühren und doppelte Scheet Nummern aufgedeckt und nachgeprüft, ob jedes Netz einen zugehörigere Netz Laber hat.

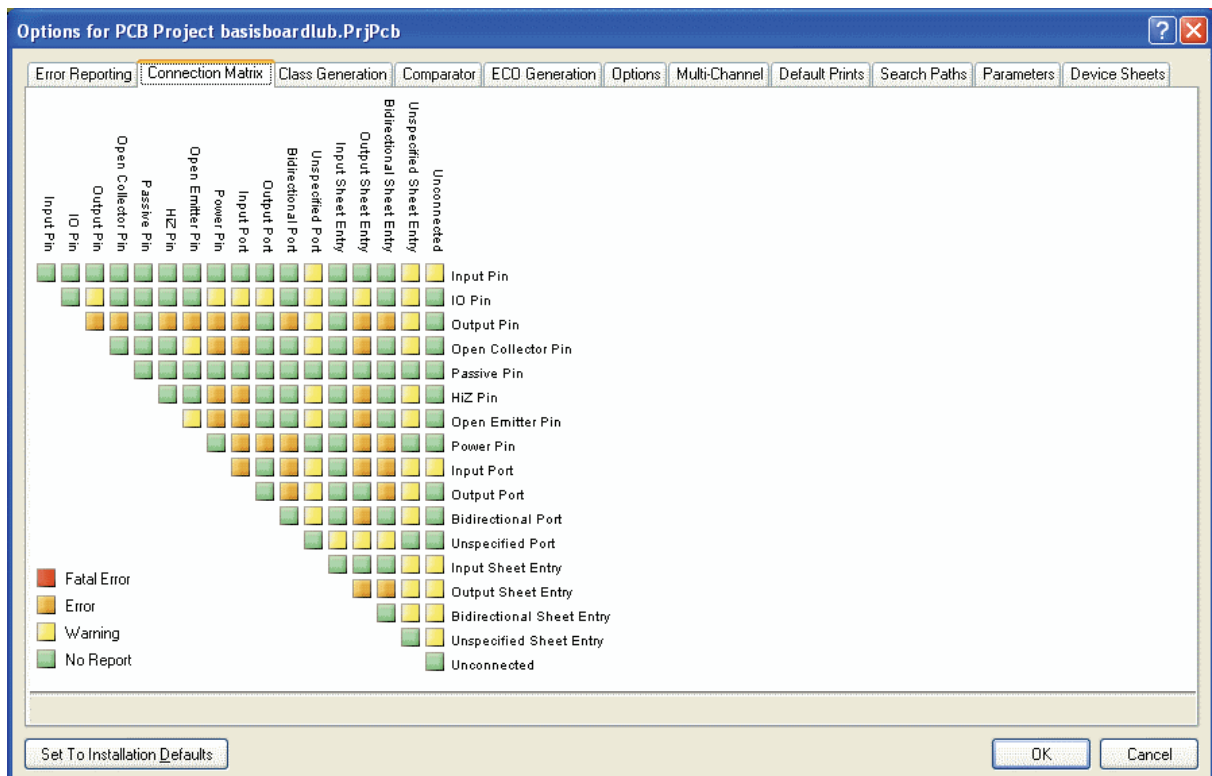
Eine elektrische Regelprüfung kann durch **Project Options** unter **Project** immer manuell in Schaltplan aufgerufen und durchgeführt werden. Es ist eine Nachbearbeitung für das spätere PCB Layout. Der Vorteil ist jedoch, dass Fehler, die vor dem PCB Layout Prozess in der ERC Phase im Entwurfsschaltplan entdeckt werden, einfacher und kostengünstiger behoben werden können. Nur nach korrekter elektrischer Regelprüfung kann ein korrektes PCB Layout produziert werden. Deshalb spielt dieser Schritt eine wichtige Rolle.

Oft gibt es zahlreiche Fehlermeldungen bei der Überprüfung der Schaltplankonstruktion. Einige davon können relevant sein, andere nicht. Manche Fehlermeldung hat überhaupt keine Bedeutung. Abbildung 6-8 lässt an Hand der **Error Reporting** Registerkarte erkennen, welche Fehlermeldungen für unseren Entwurf von Belang sind.



**Abbildung 6-8: ERC Error Reporting**

Man kann z.B die Verbindungen der PINs im Schalplan unter der zweiten Registerkarte **Connection Matrix** identifizieren, wie in Abbildung 6-9 dargestellt.

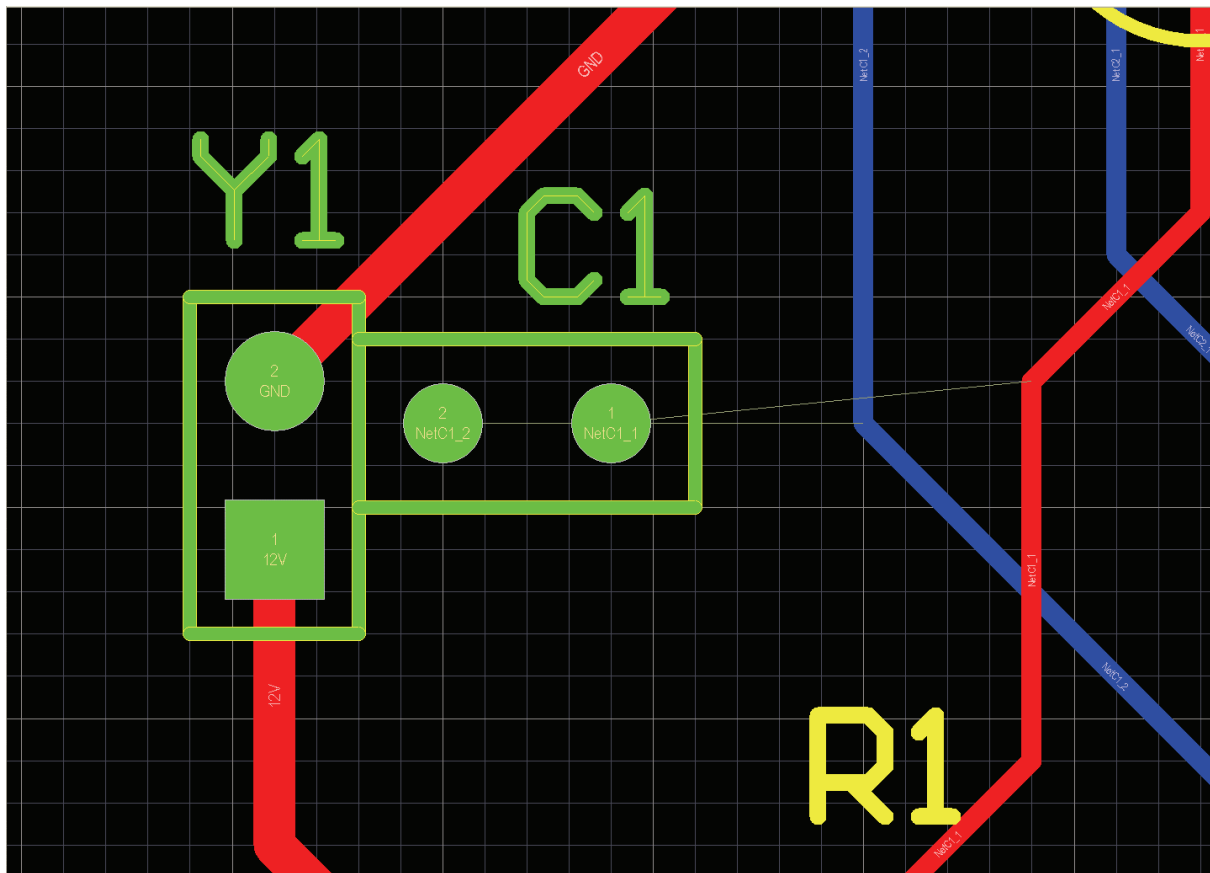


**Abbildung 6-9: ERC Connection Matrix**

## 6.5 Behebung von Komponenten-Positionsfehlern

Bei der ersten Durchführung von Design Rule Checking wurde mehr als 500 Regelverletzung identifiziert. Die meisten Fehler betreffen Regelverstöße des Mindestabstands zweier Komponenten auf der Platine. Wie in Abbildung 6-10 dargestellt, dass die Komponenten C1 und Y1 zu nahe aneinander sind. Ihr Abstand zueinander beträgt weniger als 0,3 mm (Eingabewert<sup>43</sup>) und wird in Altium Designer mit dem Highlight Grün als Fehlermeldung gekennzeichnet.

<sup>43</sup> Eingabewert für die Mindestabstand zwischen Komponenten (siehe Abschnitt 6.2.1)



**Abbildung 6-10: Mindestabstand Regelverletzung**

Von Anfang an habe ich mit Eagle<sup>44</sup> die Positionswerte der Komponenten bearbeitet. Leider wurden 90 Prozent der Komponenten bei Design Rule Checking in Altium wegen Regelverstoßes der Mindestabstände als Fehler im Message Panel angemeldet. Darum wurden erst alle Komponenten mit diesem Fehler herausgefiltert und neu positioniert.

Besonders problematisch bei dieser Art von Fehlern ist die Bus-Schnittstelle zum DSP-Modul auf dem Basisboard.

### **Potenzielle Ursache**

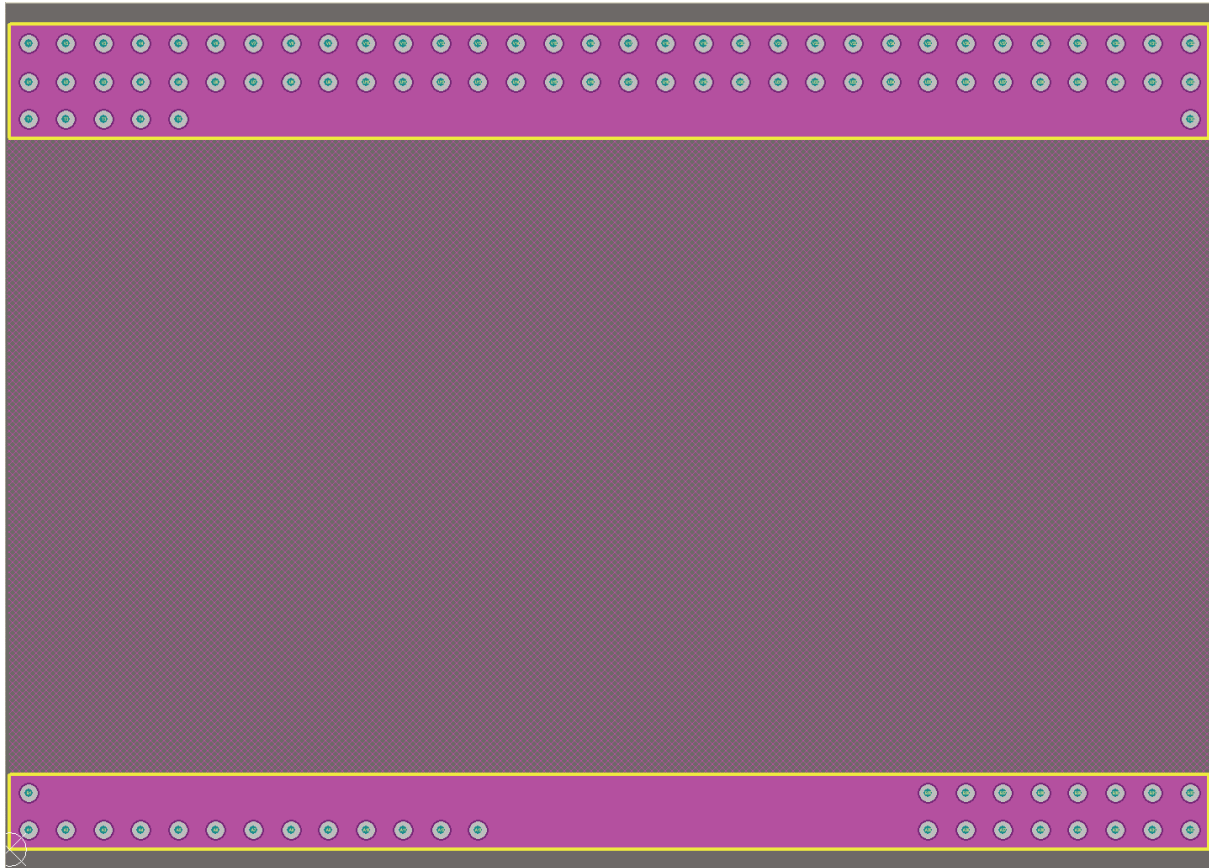
Bei der Erstellung der Bus-Schnittstelle zum DSP-Modul in der eigenen Komponentenbibliothek<sup>45</sup> wurden der obere Steckverbinder und der untere Steckverbinder als ein Bauteil erstellt. Altium Designer behandelt die beiden Steckverbinder und die zwischen legende Leerfläche als einen Komponentenkörper. Abbildung 6-11 zeigt, die im meiner Studienjahrarbeit [Dai09]

<sup>44</sup> Eagle Layout Editor von Firma CadSoft ist ein benutzerfreundliches, leitungsfähiges Werkzeug für die Entwicklung von Leiterplatten.

<sup>45</sup> In Rahmen meiner Studienjahrarbeit wurden die Komponenten, die in Altium Designer Komponentenbibliothek nicht verfügbar sind, selbst hergestellt. Dabei wurde auch eine eigene Komponentenbibliothek erzeugt.



gebaute Footprint für die Bus-Schnittstelle. Nach Positionierung dieser Bus-Schnittstelle auf der Basisboardplatine wurden alle Komponenten, die im Zwischenraum der beiden Kontaktleisten liegen, als Überlappungsfehler erkannt. Da das DSP-Modul nicht auf der Oberfläche aufliegt, werden die Bauteile auf dem Basisboard nicht beeinträchtigt.



**Abbildung 6-11: Footprints der problematischen Bus-Schnittstelle**

### **Mögliche Lösungen**

Der erste Versuch war, die Leerfläche zwischen den beiden Bus-Schnittstellen abzuschneiden. Mit dem von Altium Designer bereitgestellten Werkzeuge „Cut and Clear“ ist es nicht gelungen den Bereich vom Komponentenkörper abzutrennen.

Zwei getrennte Bauteile wären realisierbar, aber aufgrund des veränderlichen Abstands wäre es erforderlich, stets die Positionen der beiden Teilkomponenten abzugleichen. Es ist daher zu aufwendig.

Der Einsatz eines 3D Body soll folgende Funktionalität beibringen: [Alt08a]

- Ein 3D Body ist eine Polygon Design Objekt Type, das in einer Bibliothek Footprint Komponente oder einem PCB Dokument auf einem aktivierten mechanischen Layer



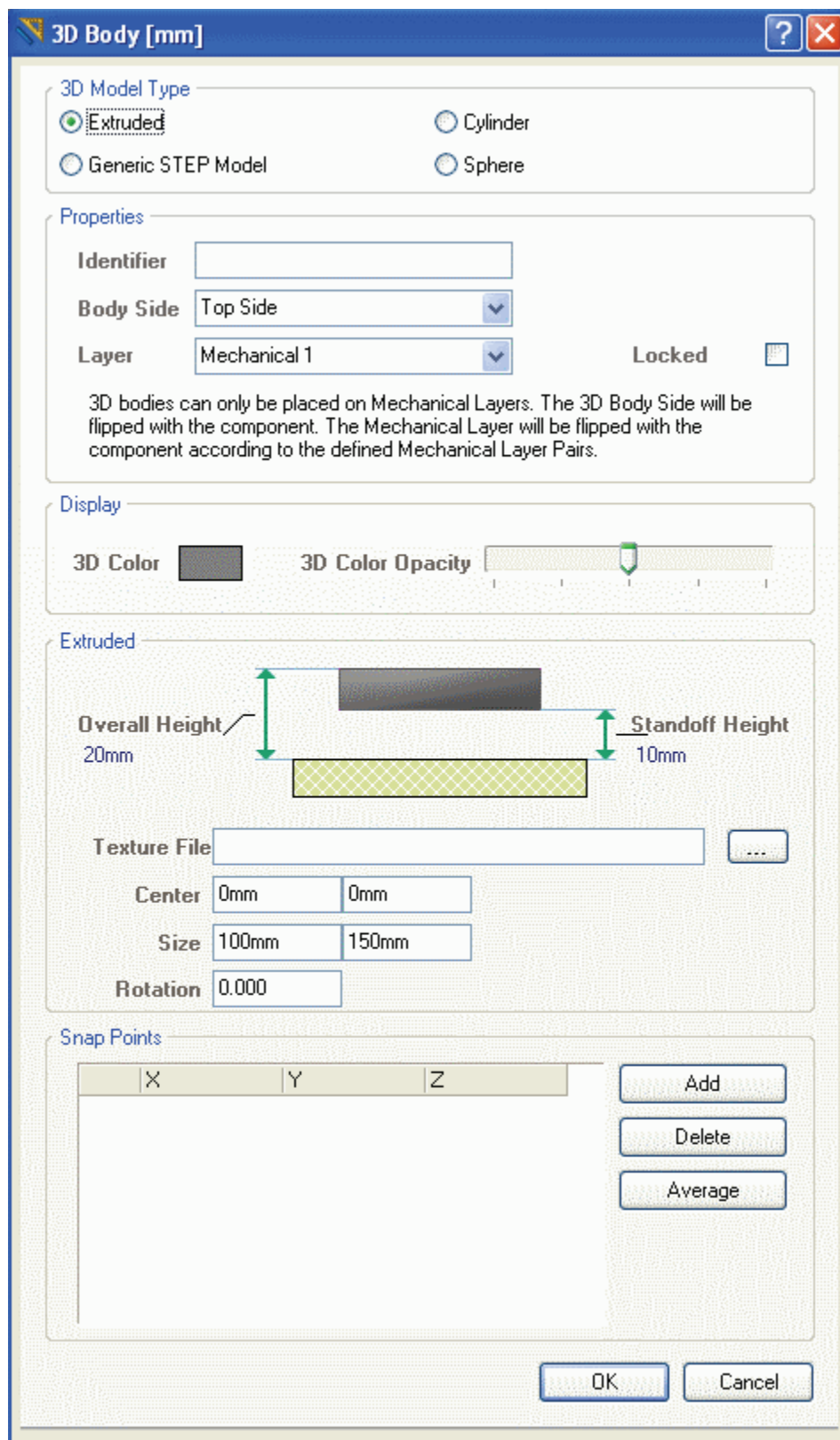
platziert werden kann. Es kann benutzt werden, um die physische Größe und Form einer Komponente speziell zu definieren, sowohl in horizontaler als auch vertikaler Ebene, und es ermöglicht eine genauere und kontrollierte Komponenten-Abstandsüberprüfung durch den Design Rule Checker.

- 3D Body verwendet den 3D Modus, um eine dreidimensionale Form für die Komponente oder das Objekt zu erstellen. Dabei dürfen mehrere individuelle 3D Bodies verwendet werden, um relativ komplexe Formen zu definieren. Dies kann in der vertikalen Ebene besonders nützlich sein, da es erlaubt, die Höhe einer Komponente in den verschiedenen Regionen zu variieren.
- Das 3D Body Objekt dient auch als ein Platzhalter für eingebettete oder verknüpfte 3D STEP Modell<sup>46</sup> Dateien.

Hier interessiert nur die zweite Eigenschaft. Für die Leerfläche kann dazu ein 3D Body eingesetzt und auf die Höhe dieser Fläche gestellt werden. Damit wird ein Zwischenraum erstellt, in dem die unteren Bauelemente platziert werden können. Für den Einsatz eines 3D Body stehen der PCB und der PCB-Bibliothek Editors zur Verfügung. Er kann auch manuell durch Auswahl von **Place \ 3D Body** eingefügt werden.

---

<sup>46</sup> Viele Komponenten Hersteller liefern detaillierte 3D Modelle für die Verwendung in mechanischen CAD Paketen. Altium Designer können 3D STEP Modelle importieren STEP direkt als 3D body Objekt



### Abbildung 6-12: 3D Body

Nach dem Startbefehl erscheint der 3D Body Dialog, wie in Abbildung 6-12 zu sehen ist. Unter **Properites** kann eine Region definiert werden, die dann extrudiert wird. Dazu wählt man die **Extruded** Option in **3D Modell Type** Region. Außerdem kann ein Name vergeben und

---

Inventarisierungsnummer: 2010-02-24/019/IN01/2231 59

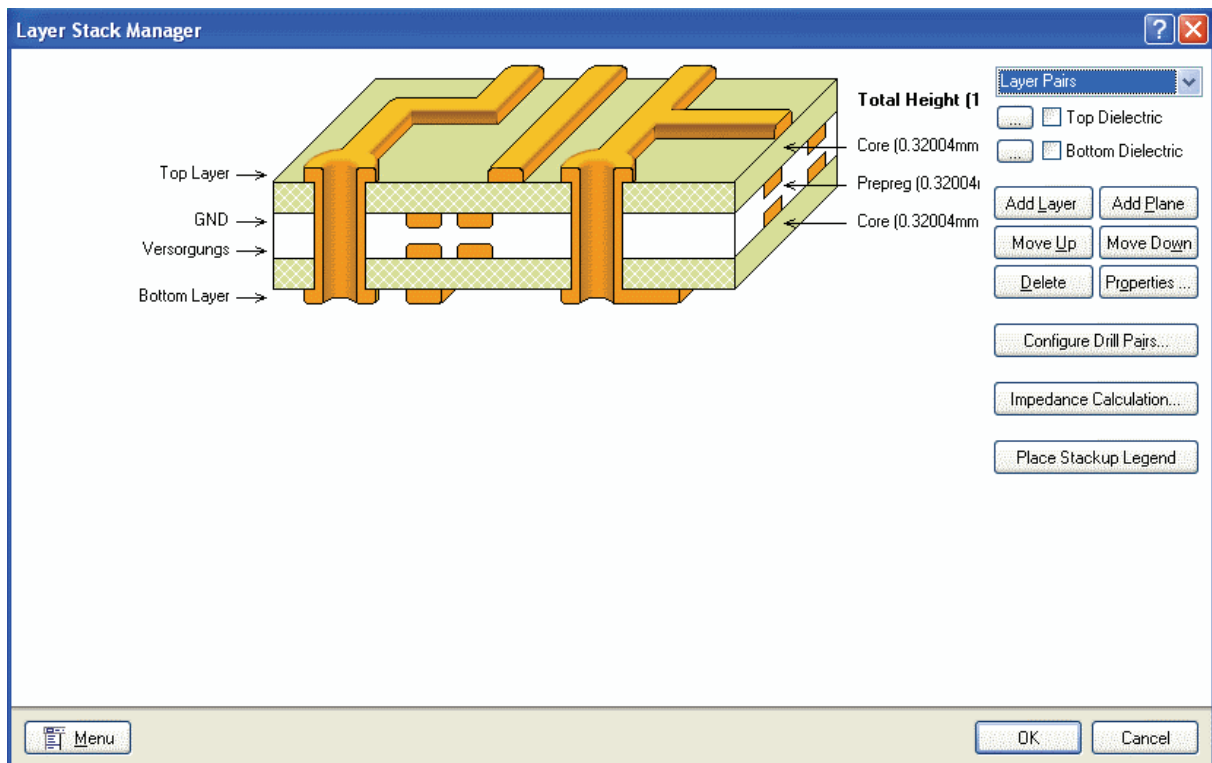
festgelegt werden, auf welcher Lage sie sich befinden soll. Danach kann die Gesamthöhe (Overall Height), das ist der Abstand zwischen der Oberfläche des Basisboards zur Oberfläche des 3D Bodys, festgelegt werden. Für das beschriebene Bauteil wurde eine Overall Height von 20 mm festgelegt. Optional kann die Standoff Height definiert werden, dies ist der Abstand zwischen der Oberfläche des Basisboards und der Unterseite des 3D Bodys. In diesem Fall wird eine Standoff Height von 10 mm verwendet.

Nach Schließen des 3D Body Dialogs befinden wir uns im Platzierungsmodus. Die gewünschte Platzierung der Bus-Schnittstellen konnte erfolgreich durchgeführt und damit das eigentliche Problem behoben werden.

## **6.6 Hinzufügen zusätzlicher Layer**

Ein einfacher Entwurf kann nur auf einer einseitigen oder doppelseitigen Platine verdrahtet werden. Ist das Design komplexer, so können mit Hilfe des Layer Stack Manager Dialog mehrere Schichten hinzugefügt werden.

Bei diesem Entwurf verlangt unsere Leitplatte vier Layer-Arten: Top Layer, GND- Layer, Versorgungs- (VCC) Layer und Bottom Layer. Die zusätzlichen GND- und VCC-Layer müssen extra einer bestehenden Platine hinzugefügt werden. Klickpfad: **Design \ Layer Stack Manager** (siehe Abbildung 6-13).



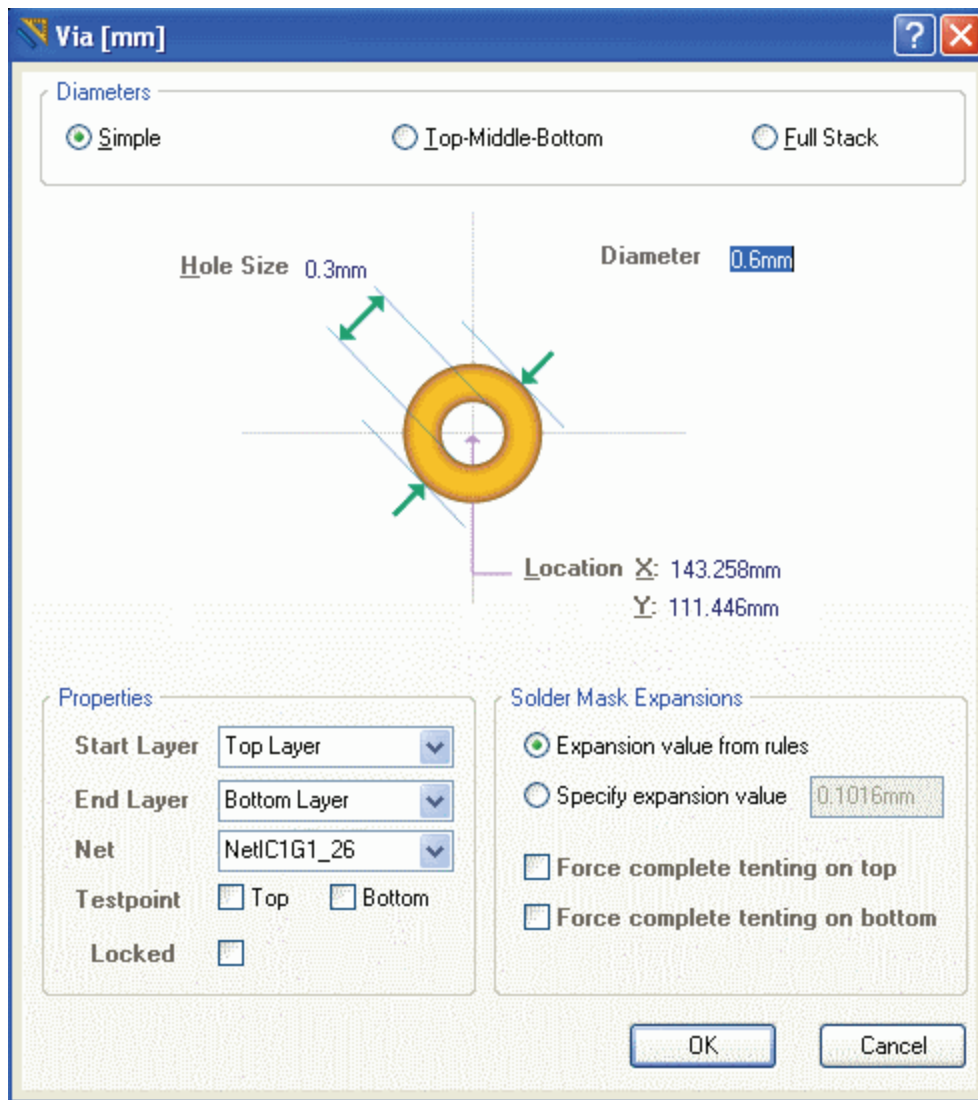
**Abbildung 6-13: Layer Stack Manager**

Durch Auswählen einer vorhandenen Schicht und Betätigen der Schaltfläche **Add Layer** oder **Add Plane** können neue Signale bzw. eine Plane-Schicht in den Entwurf eingefügt werden. Die neue Schicht wird unter der aktuell ausgewählten Ebene hinzugefügt. Sobald die geforderten Schichten ergänzt wurden, verwendet man **Move up** und **Move down**, um den Layer Stack zu konfigurieren. Durch Doppelklicken auf einen der Layer-Namen können die Eigenschaften dieser Schicht bearbeitet werden.

## 6.7 Einsatz von Vias

Ein Via ist ein einfaches Design Objekt. Es wird verwendet, um eine elektrische Verbindung zwischen zwei Signal-Layern einer Leiterplatte zu bilden. Vias ähneln runden, durchbohrten Pads, welche in der Regel auch vernickelt werden, wenn die Leiterplatte hergestellt wird.

Durch Betätigen der Schaltflächen **Place \ Via** aus dem PCB Editor Hauptmenü können Vias gesetzt werden, ebenso durch Klicken auf den Button auf der Verdrahtung Toolbar. Nun wird der Cursor zur gewünschten Stelle bewegt und mittels Eingabetaste bestätigt, um das Via zu platzieren. Abbildung 6-14 veranschaulicht die Bearbeitung eines Vias im Dialog Via Eigenschaften. Durch Drücken der TAB-Taste während der Positionierung kann der Dialog Via Eigenschaften aufgerufen werden.



**Abbildung 6-14: Via Eigenschaft**

In diesem Dialog können der innere und äußere Durchmesser des Via eingegeben werden, Ferner wird erfasst, bis zu welchem Layer Via auf die Platine gebohrt werden und mit welchem Netz sie in Verbindung stehen.

## **7 Altium Designer Routing Technologie**

Nach der Platzierung der Bauelemente, dem Einsatz von Vias und der Einstellung von Regeln folgt nun die wichtigste Phase für den PCB-Entwurfsprozess – das Routing der Verbindungen. Der Altium Designer enthält eine Reihe intuitiver interaktiver Routing Funktionen, deren Aufgabe es ist, eine einfache beidseitige Leiterplatte effizient und genau zu einer Multilayer-Leiterplatte mit hoher Dichte und Geschwindigkeit zu verdrahten.

In diesem Kapitel werden die begrenzten Möglichkeiten derzeitiger Routing Technologien unter geometrischen Bedingungen betrachtet und neue Komponenten und Technologien zur Leiterplattenkonstruktion diskutiert.

### **7.1 Autorouter in Altium Designer**

Das Routen von Verbindungen auf einer Leiterplatte ist ein komplexer und zeitaufwändiger Vorgang. Bei großen oder dichten Leiterplatten kann dies eine beträchtliche Zeit in Anspruch nehmen - ein Faktor, der bei den immer kürzeren Lebenszyklen der Produkte zunehmend an Bedeutung gewinnt. Autorouter unterstützen den Konstrukteur, indem sie beim Verlegen von Leiterbahnen Tracks und Vias automatisch auf der Leiterplatte platzieren. Sie erleichtern somit diesen intensiven und komplexen Vorgang.

#### **7.1.1 Probleme des Autorouting**

Ein elektronisches Design besteht aus zahlreichen Komponenten, die auf einer mechanischen Struktur (Leiterplatte) mit mehreren Ebenen implementiert und deren Pins auf spezielle Weise miteinander verbunden sind. Diese physischen Verbindungen bestehen aus Kupferbahnen (Route), welche auf der und durch die Leiterplatte hindurch verlaufen und die Pins miteinander verbinden.

Die Schaffung der Kupferbahnen (Routen) zwischen den Pins ist ein komplexer und komplizierter Vorgang, da mitunter tausende Verbindungen auf engstem Raum zwischen den Komponenten erstellt und in hoher Dichte auf der Leiterplatte platziert werden müssen.

Obwohl die meisten Hersteller moderner Autorouter dieses Problem bis zu einem gewissen Grade meistern, erreichen Grid based, Shape based oder geometrische Methoden beim Abbilden der Leiterbahnfläche hier ihre Grenzen - und dies ausgerechnet bei den in der Leiterplattenherstellung immer öfter zum Einsatz kommenden Pakettechnologien für dichtere, nicht orthogonale und geometrisch unregelmäßige Komponenten.

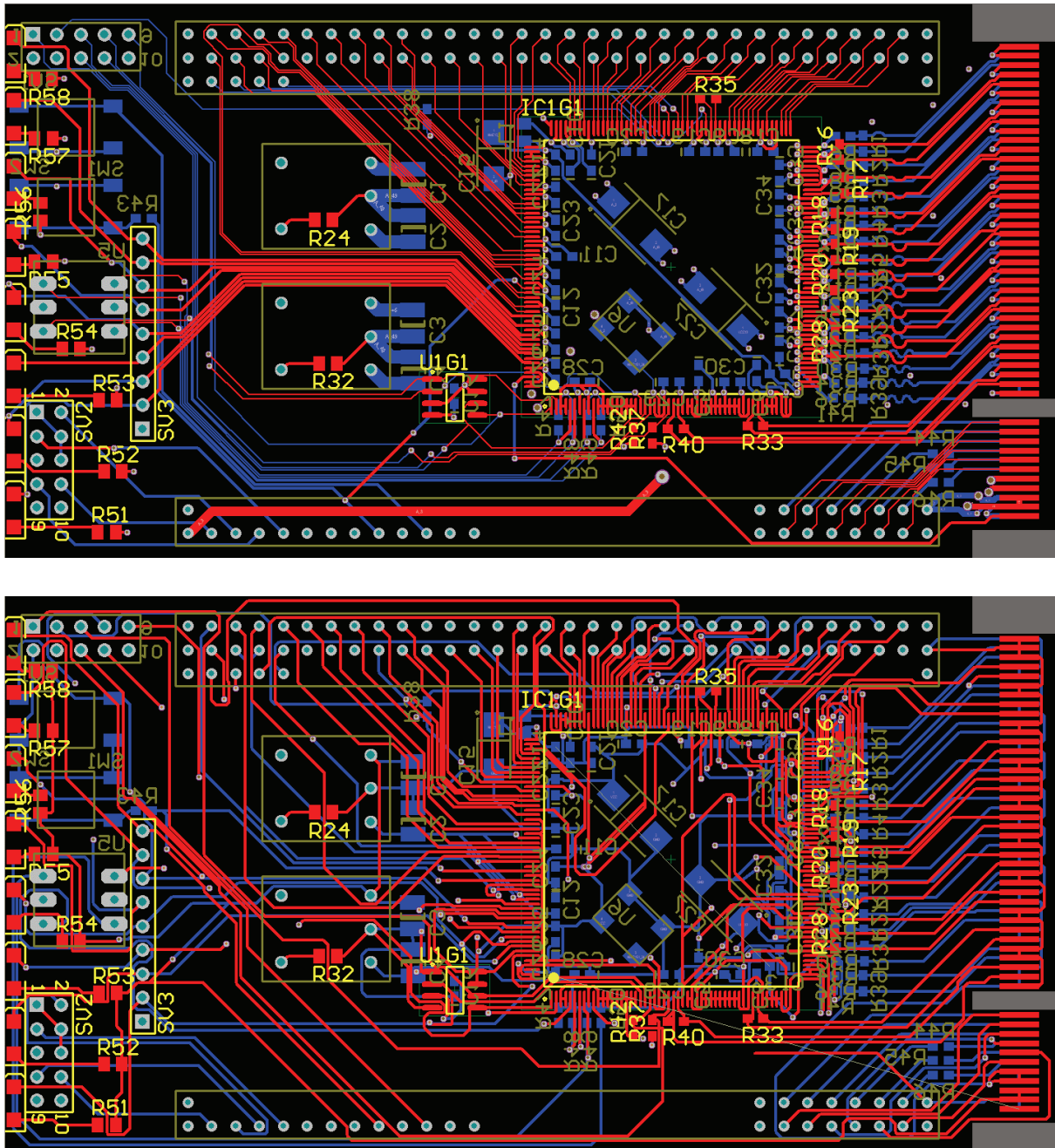
Erschwerend kommt hinzu, dass die Autorouter der aktuellen Generation auf Grund geometrischer Beschränkungen Ergebnisse liefern, die mit hohem Aufwand manuell nachbearbeitet werden müssen.

Keinerlei derartigen Einschränkungen unterliegt hingegen der von Altium-Limited entwickelte Situs Autorouter. Zur Abbildung der Leiterplattenoberfläche kommt hier ein topologisches Analyseverfahren zum Einsatz. Durch topologisches Mapping wird eine größere Flexibilität bei der Bestimmung der Leitwegpfade erreicht, weil, anders als beim geometrischen oder Shape based Mapping Obstacle, die Berücksichtigung von Formen oder Koordinaten entfällt. Zudem unterliegt die Routing-Richtung keinerlei Einschränkungen.

Auf Grund der wesentlichen Vorteile des topologischen Autoroutings wird das Altium-Designer-Werkzeug für das Basisboard unserer Entwurfsplatine nur zur Probe verwendet. Das Autorouting von Altium Designer hat in 14 Minuten versucht, alle Leitungen zu verdrahten. Leider ist Endergebnis nicht optimal.

- Erstens: Nicht alle Leitungen wurden verdrahtet
- Zweitens: Die Verbindungen, die sich auf GND-Layer und VCC-Layer befinden sollten, wurden nicht berücksichtigt. Stattdessen wurden diese Leitungen auf dem Top- oder Bottom Layer berücksichtigt.
- Außerdem sind die Leitungen auf dem Top- und Bottom Layer ziemlich ungeordnet. Dies könnte die spätere Anwendung der Interactive Length Tuning Funktion erschweren.

Abbildung 7-1 vergleicht das manuell per Hand verdrahtete Basisboard mit der von Altium Designer Autorouting verdrahteten Basisboard Platine. Oben ist die per Hand verdrahtete Platine zu sehen, unten die Platine, die von Altium Designer Autorouting verdrahtet wurde.



**Abbildung 7-1: Manuell Routing (oben) und Autorouting (unten) in Vergleich**

### 7.1.2 Fazit des Autorouting

Das topologische Autorouting von Altium Designer stellt für diesen Entwurf keine ideale Lösung dar. Tatsächlich scheuen viele Konstrukteure allein wegen der genannten Einschränkungen vor dem Einsatz von Autoroutern zurück. In diesem Entwurf wurde auf Autorouting komplett verzichtet. Alle Verbindungen wurden mit dem Tool Interactive Routing manuell per Hand hergestellt.



## 7.2 Interactive Routing Funktion

Im Anschluss werden die beiden Leitplatten geroutet. Das heißt, dass den bisher symbolischen Leiterzügen konkrete Positionen einschließlich Vias zugeordnet werden. Nach Auswahl des gewünschten Layers gelangt man über das markierte Symbol zum interaktiven Routing (Verdrahtung). Nach Auswahl dieses Punktes können die Leiterzüge mit der Maus verlegt werden. Dabei kann der Layer während des Routens über die Taste \* des Ziffernblocks gewechselt werden. Die Taste Space wechselt den Vorzugswinkel der Leitungen beim Routen. Der Modus wird mit der Taste ESC oder der rechten Maustaste beendet.

### 7.2.1 Eingabe einer nicht vordefinierten Netzbreite

Für die letzte Kontrollstufe sowie für die Leitungsbreite, die in Design Rules nicht vordefiniert ist, kann man während der Routingprozesses jederzeit beliebige Routingbreiten eingeben. Durch Drücken der TAB Taste öffnet sich der **Interactive Routing for Net** Dialog, wie in Abbildung 7-2 dargestellt.

Hier können noch einmal genaue Werte für Länge, Breite oder Via-Größe eingegeben werden.

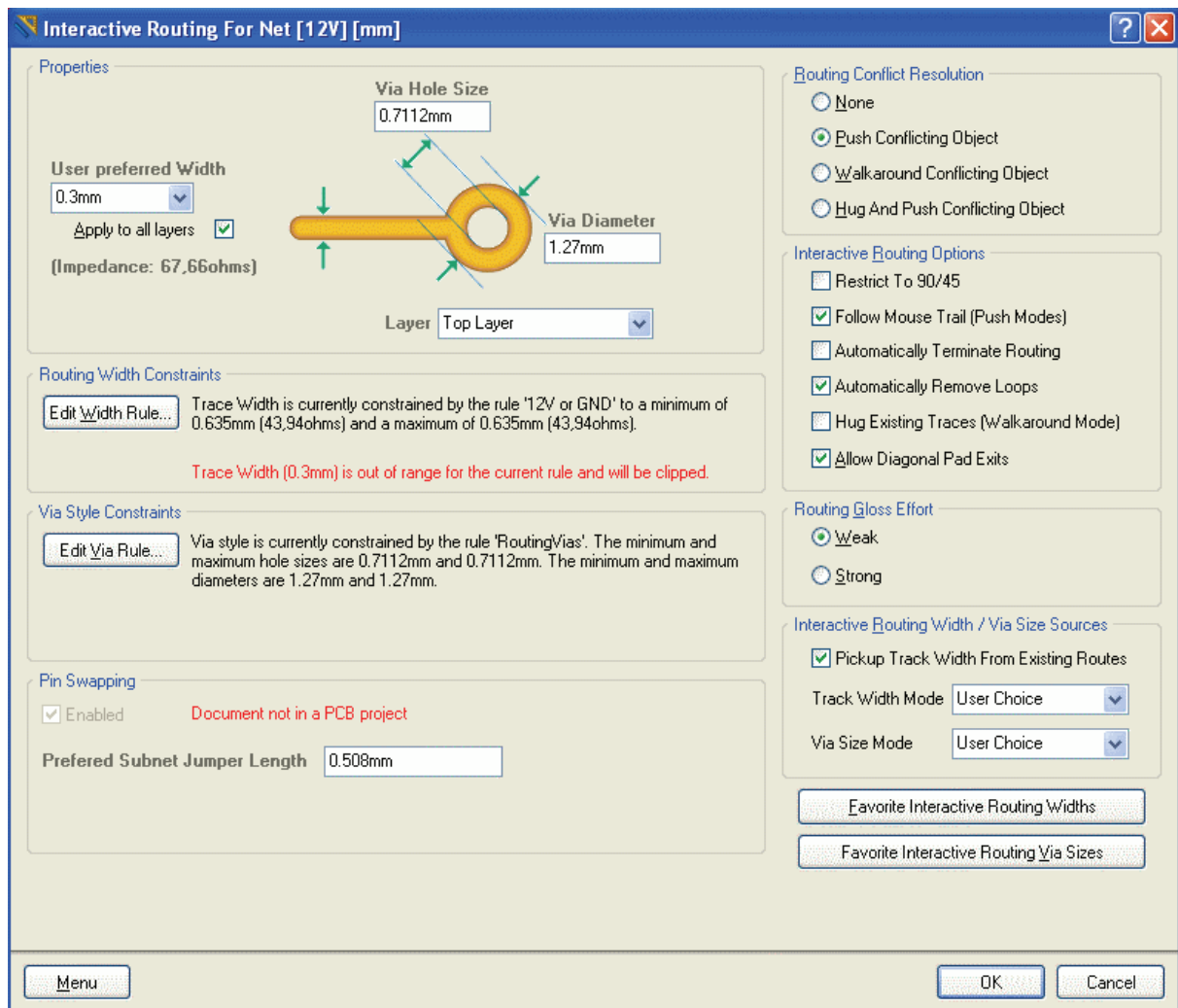


Abbildung 7-2: Interactive Routing for Net

## 7.2.2 Smart Interactive Routing

Mit der Taste STRG und dem gleichzeitigen Anklicken eines Bauteiles lassen sich Routing-Pfade und entsprechende Leiterzüge mittels Altium Designer auch manuell analysieren und bestimmen – eine Kombination aus manuellem Routing und Autorouting, wie in [Alt06] erläutert.

Die Erfahrung hat beispielsweise gezeigt, dass es am effizientesten ist, alle Verbindungen mit einer gemeinsamen Schicht in dieselbe Richtung zu routen. Bei einfachen Leiterplatten mit zwei Schichten werden dabei eine horizontale und eine vertikale Schicht bestimmt. Sowohl der Konstrukteur als auch der Router können dann die Routen entsprechend dieser Festlegung platzieren.

Wie die Spuren auf einer Autobahn bringt dieser Ansatz Ordnung in den Routing-Vorgang und ermöglicht dem Konstrukteur oder Router, die Leiterplatte als Reihe von Kanälen zu betrachten, die dann geregelt zugewiesen werden können.

### 7.3 Interactive Length Tuning

Die optimale Nutzung des Potentials der neuesten, hochdichten High Speed Bauteile, wie z.B. der neuesten Generation von programmierbaren Bausteinen, stellt für die Design Werkzeuge insbesondere im Routing Bereich eine echte Herausforderung dar. Altium Designer hat diesen Bedarf erkannt und die bereits sehr leistungsstarke Auswahl von interaktiven Routing Werkzeugen im Altium Designer Version 6 noch erweitert. Hinzugekommen sind neue Funktionen für High Speed Designs, wie etwa das intelligente „Interactive Length Tuning“.

Die interaktive Length Tuning Funktion ergänzt die vorhandenen High-Speed, High-Density Board<sup>47</sup> Layout Funktionen, das impedanzkontrollierte Routing, interaktives Differential Paar Routing, BGA Escape Routing sowie die integrierte Signalintegritätsanalyse mit Assistent für Leitungsabschlüsse und viele andere. Die Length Tuning Funktion lässt sich nahtlos mit diesen Fähigkeiten kombinieren und stellt ihnen eine umfassende, interaktive Lösung speziell für die High Speed, High Density Board Designs von heute zur Verfügung.

Die Anpassung der Leiterbahnlänge ist eine Standardmethode zur Sicherstellung der Datenintegrität in einem digitalen High Speed System und ein zentraler Bestandteil des Differential Pair Routing. Hierbei optimieren wir nicht angepasste Netzlängen, indem wir die Länge von entflochtenen Netzen reduzieren oder verlängern.

Interactive Length Tuning bietet eine dynamische Möglichkeit zur Optimierung und Steuerung der Netzlängen, indem variable Amplitudenmuster (oder „Akkordeon“-Segmente) entsprechend dem verfügbaren Platz, den Regeln oder den Hindernissen in das Design eingefügt werden. In diesen Entwurf verlangen alle LVDS Verbindungen, wie in Kapitel 4 unter Abschnitt 4.1 schon erwähnt, gleiche Netzlängen.

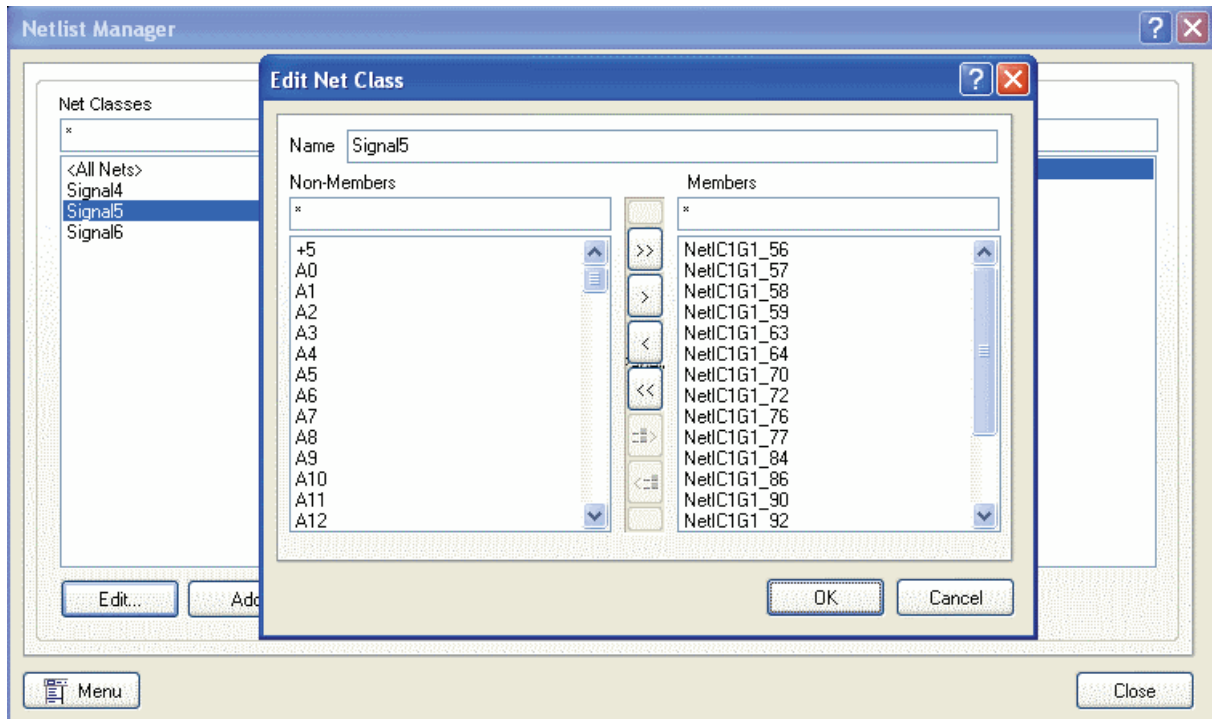
#### 7.3.1 Netzklasse

Vor der Anpassung der Netzlängen müssen zunächst die entsprechenden LVDS-Leitungen ausgewählt und zu einer Netzklasse zusammengefasst werden. Mit Hilfe des Untermenüs **De-**

---

<sup>47</sup> Hohe Geschwindigkeit und hochverdichtete Leiterplatten sind neue Konstruktionsstrategien, um entweder mehr Funktionen auf kleineren Platten oder mehr Funktionen auf größeren Platten zu bewältigen.

**sign \ Netlist \ Edit Nets** wird der Netlist Manager Dialog (siehe Abbildung 7-3) angezeigt. Bei dem in der Abbildung dargestellten Beispiel wird aus den Listen per Eingabe eine neue Netzkategorie mit Name und Auswahl der zugehörigen Leitungen erstellt.

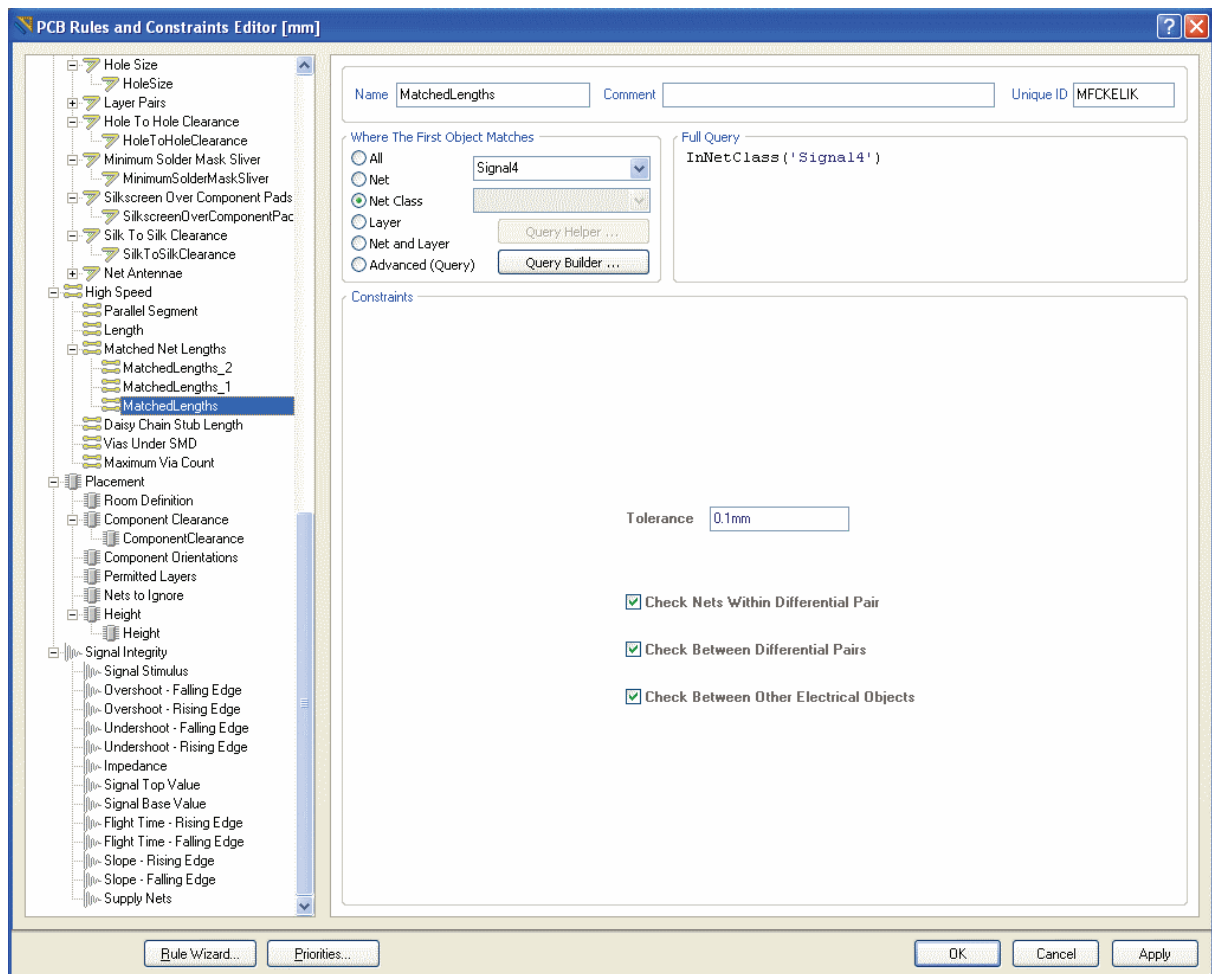


**Abbildung 7-3: Netz Manager**

### 7.3.2 High Speed Rules

Für die erstellte Netzkategorie müssen nun noch die passenden High Speed Rules eingegeben werden. Die **Matched Net Lengths** Design-Regel unter **High Speed** im Dialog **PCB Rules and Constraints Editor** bestimmt, dass Leitungen mit gleicher Länge innerhalb der vorgegebenen Toleranz zu verdrahten sind.

InNetClass ('MyEqualLengthNets') bedeutet, dass die Regel für alle Leitungen der Netz Klasse MyEqualLengthNets gilt. Hier kann der künftige Name der Netzkategorie eingegeben werden.



**Abbildung 7-4: High Speed Rule**

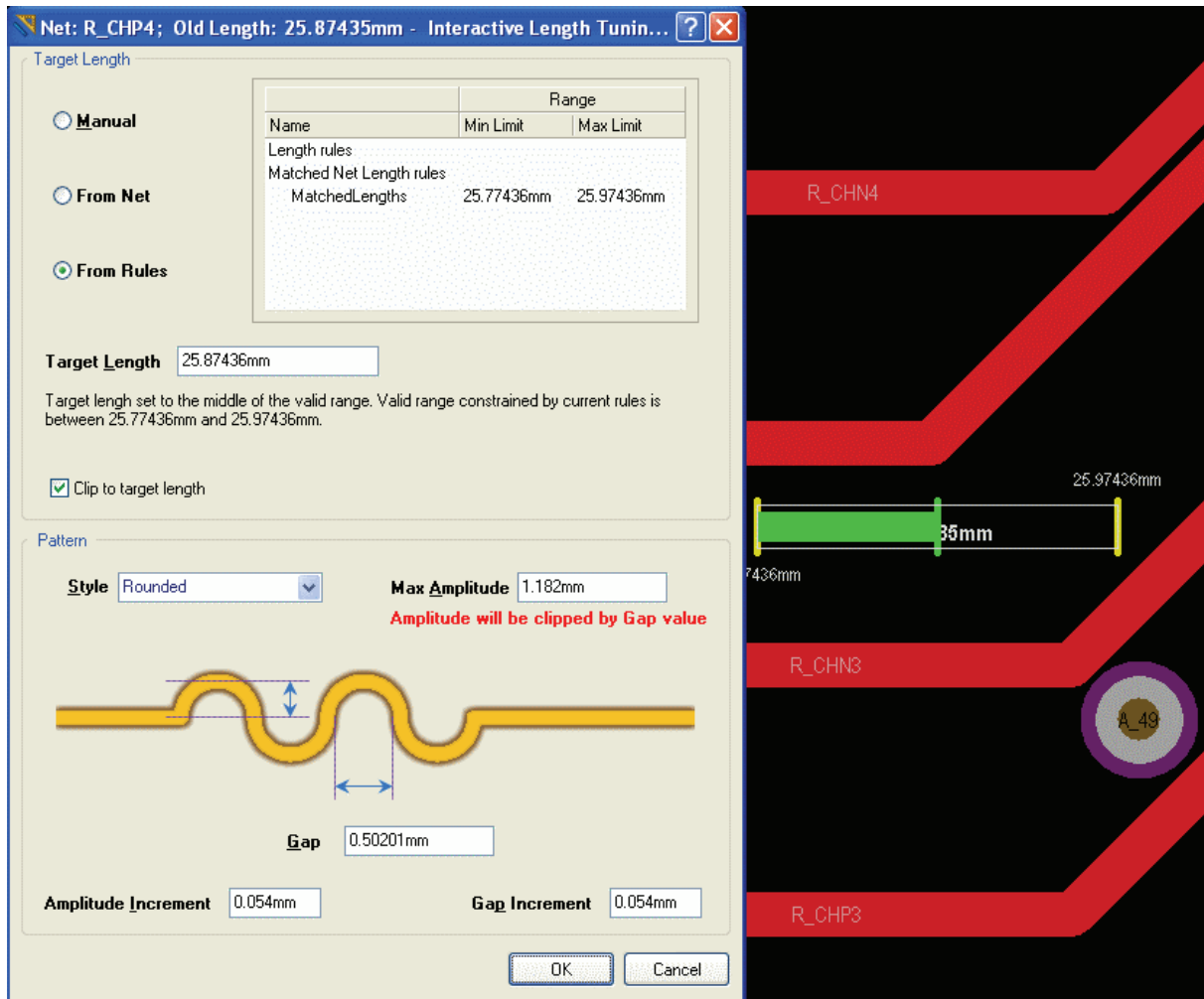
Abbildung 7-4 zeigt, dass die Toleranz für diesen Entwurf auf 0.1 mm festgelegt wurde, um eine minimale Abweichung bei gleicher Netzlänge zu erreichen.

## 7.4 Interactive Length Tuning Routing Funktion

Diese Funktion wird vom Menü aus aufgerufen und kann auf Designregeln oder Netzeigenschaften basieren oder manuell aus den Werten bestimmt werden, die in ein Dialogfeld eingegeben werden. Length Tuning kann mit geraden oder gebogenen Segmenten durchgeführt werden, wobei Designer die volle Kontrolle über Amplitude, Abstand und Eckradius hat.

Nach dem üblichen Routing und der Eingabe des Befehls erfolgt das Hinzufügen der Tuning-Segmente durch Anklicken des entflochtenen Netzes und Bewegen der Maus entlang des Netzes. Der Interactive Length Tuning Cursor führt uns durch den Tuning Prozess. Die gelben Cursor Balken zeigen die möglichen minimalen und maximalen Längen. Der grüne Balken zeigt die Ziellänge entsprechend den jeweiligen Matched Length und Max Length Design Regeln oder den Einstellungen im Interactive Length Tuning Dialogfeld. Der bewegliche An-

zeiger gibt an, wie nahe wir einer Übereinstimmung sind (siehe Abbildung 8-5). Während des Length Tuning stehen komfortable Tastenabkürzungen zur Kontrolle von Stil und Format des Tunings zur Verfügung. So dienen z.B. die Tasten 3 und 4 zur Verkleinerung oder Vergrößerung des Neigungswinkels.



**Abbildung 7-5: Interactive Length Tuning**

Wichtig ist an dieser Stelle der Hinweis, dass sich das Interactive Length Tuning an Abstands-Designregeln hält, so dass die Platzierung von Objekten auf anderen Netzen vermieden wird. Abbildung 7-5 stellt den Zugriff auf das Length Tuning Dialogfeld durch betätigen der TAB Taste während des Tunings dar.

Laut [Alt08a] gehören zu den wichtigsten Funktionen des Interactive Length Tuning Werkzeugs gehören:

- Einheitliche Oberfläche: Wie bei den anderen Altium Designer Interactive Routing Werkzeugen gewährleistet sie eine schnelle und bequeme Bedienung an Hand bereits bekannter Tastaturabkürzungen.
- Die Target Length kann entweder durch Design Regeln, durch ein bestehendes Netz oder manuell festgelegt werden.
- Es besteht die Möglichkeit, die Tuning Muster direkt an die Target Length anzupassen, wenn „Mitered with Lines“ und „Mitered with Arcs“ benutzt werden.
- Es stehen drei Tuningarten zur Verfügung: „Mitered with Lines“, „Mitered with Arcs“ und „Rounded“. In unserem Projekt werden alle Tuningarten auf „Rounded“ gesetzt.

Trotz des professionellen Einsatzes der Interactive Length Tuning Funktion wurde beim Versuch, die gleiche Netzlänge für LVDS-Leitungen auf dem Bottom Layer des Basisboards anzupassen, ein Problem deutlich, welches sich nur durch ein Redesign eines Teils der Basisboard Platine lösen lies. Dieses Problem und seine Lösung wird im folgenden Abschnitt detailliert erläutert.

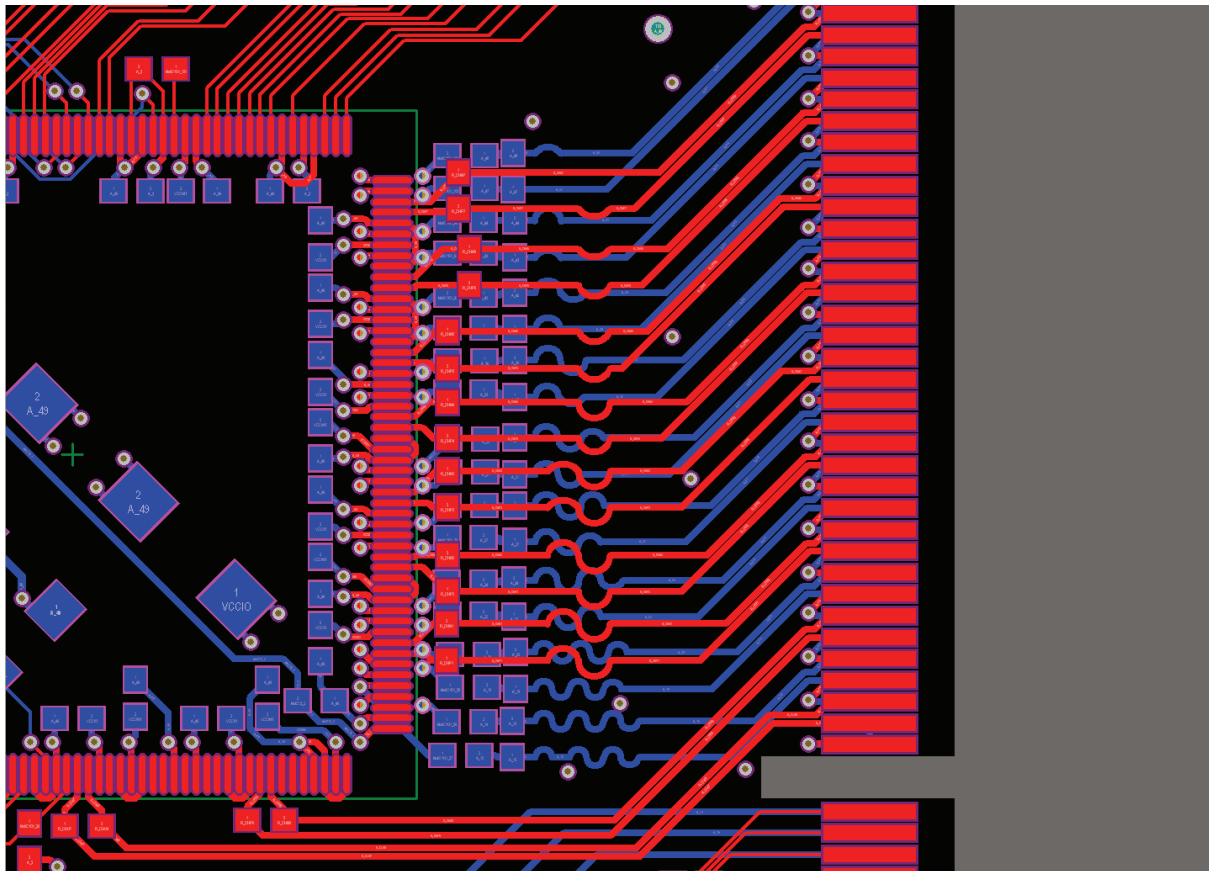
#### **7.4.1 Betroffene Problem bei Interactive Length Tuning**

Wie oben schon angedeutet, ergab sich bei dem Versuch, die Leitungen unter Verwendung der Interactive Length Tuning Funktion gleich lange zu bekommen, plötzlich ein Problem. Es sei an dieser Stelle darauf hingewiesen, dass dieses Problem vorher nicht bekannt war. Aus diesem Grund gestaltete sich die Fehlerbeseitigung keineswegs einfach und nahm vergleichsweise viel Zeit und Geduld in Anspruch. Dieser Abschnitt nennt die aufgetretenen Komplikationen und zeigt auf, wie sie behoben bzw. umgangen wurden.

##### **Ursache des Problems**

Jede LVDS Leitung auf dem Bottom Layer des Basisboards beinhaltet zwei Widerstände und ein Via (Durchkontaktierung) vom Bottom Layer zum Top Layer und dort die Verbindung zum FPGA auf dem Top Layer. Daher sind zwei Netznamen notwendig, um eine Verbindung zu beschreiben. Für die Angleichung der Leitungslängen kann eine Regel definiert werden, die dann auf eine Netzklasse angewendet wird. Dies funktioniert daher nur für die einzelnen

Abschnitte, und erfordert das zweifache Optimieren aller Abschnitte jeweils für die abschnittsweise beteiligten Leitungen (siehe folgende Abbildung 7-6).



**Abbildung 7-6: Der Problembereich: Anpassung der Leitungen auf gleiche Länge**  
**Ausgangslösung**

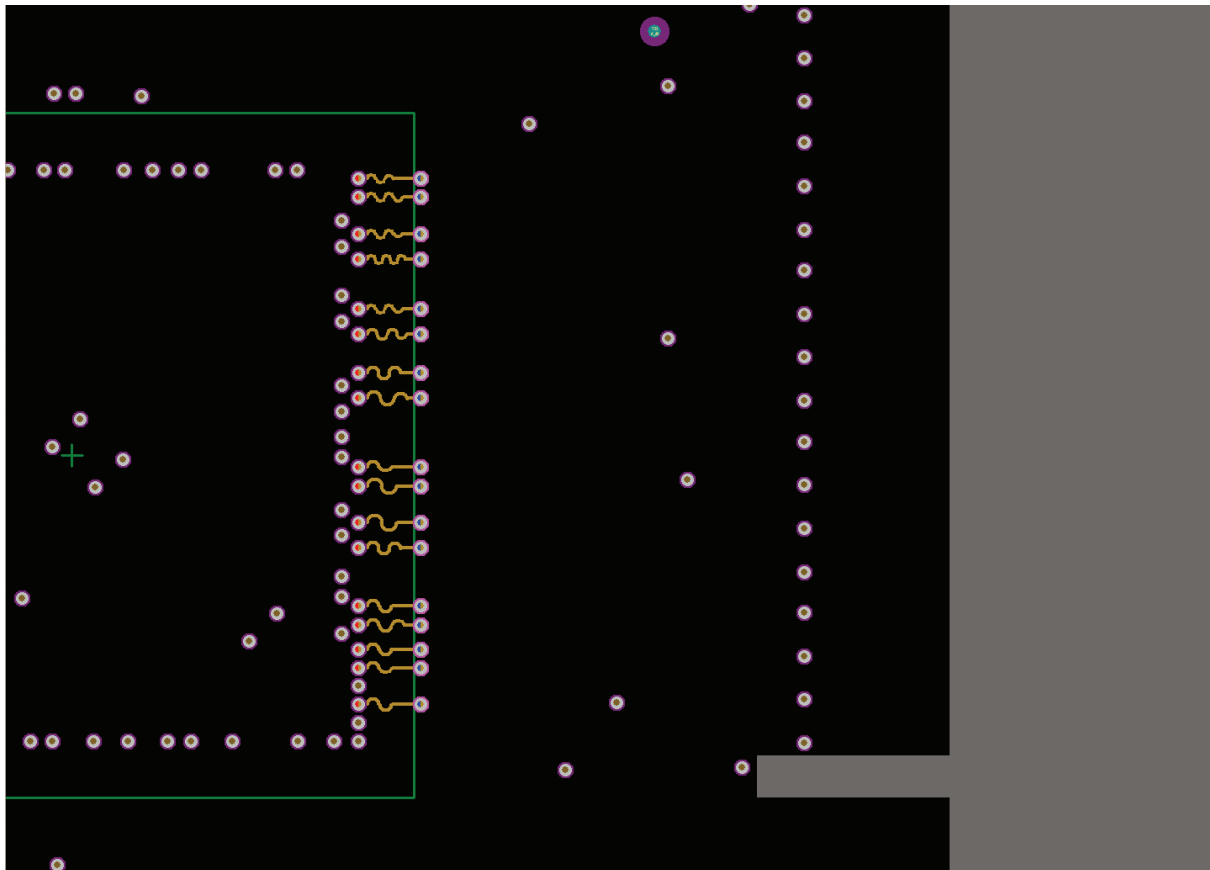
Zur Beschreibung verwenden wir zwei getrennte Netzklassen mit den Namen Signal 5 und Signal 6, Netzkategorie Signal 6 bezeichnet die Leitung vom Edge Connector zum ersten Widerstand und Netzkategorie Signal 5 die Leitung vom zweiten Widerstand zum FPGA. Hier tritt aber zweites Problem auf.

Die Leitung NetIC1G1\_57 auf dem Bottom Layer ist viel länger. Es ergibt sich keine Möglichkeit die restlichen Leitungen in der selben Netzkategorie 5 auf gleiche Länge anzupassen, da entweder auf dem Top Layer oder auf dem Bottom Layer des Basisboards die Bauelemente zu dicht beieinander platziert sind. Es ist daher kein Platz für die gebogenen Leitungen (siehe Abbildung 7-6).



## Lösungsidee

Die Idee besteht darin, einige zusätzliche Umleitungen auf dem anderen Layer einzufügen, um dort die Netze verlängern zu können. Auf dem GND Layer und dem Versorgungs-Layer ist der erforderliche Platz verfügbar. Bei diesem Projekt wurden alle Umleitungen auf den GND Layer verlegt, wie in Abbildung 7-7 demonstriert.



**Abbildung 7-7: Umleitungen auf GND Layer**

Um den GND Layer zu erreichen benötigt man noch zusätzlichen Vias. Es wurden genau 17 zusätzliche Vias für die restlichen 17 Leitungen (außer Leitung NetIC1\_57) eingesetzt. Alle zusätzlichen Vias sind auf dem Top Layer mit FPGA verbunden und nur bis zum GND Layer durchkontaktiert. Der Einsatz dieser 17 zusätzlichen Vias machte es erforderlich, viele Komponenten neu zu positionieren, besonders die Kondensatoren auf der rechten Seite unter dem FPGA. Auch die vielen GND VCC Vias kamen auf neue Positionen, usw. Die ursprünglichen Vias, die eine Durchkontaktierung vom zweiten Widerstand auf dem Bottom Layer auf die FPGA auf dem Top Layer darstellten, wurden auch nur zum GND Layer geführt.

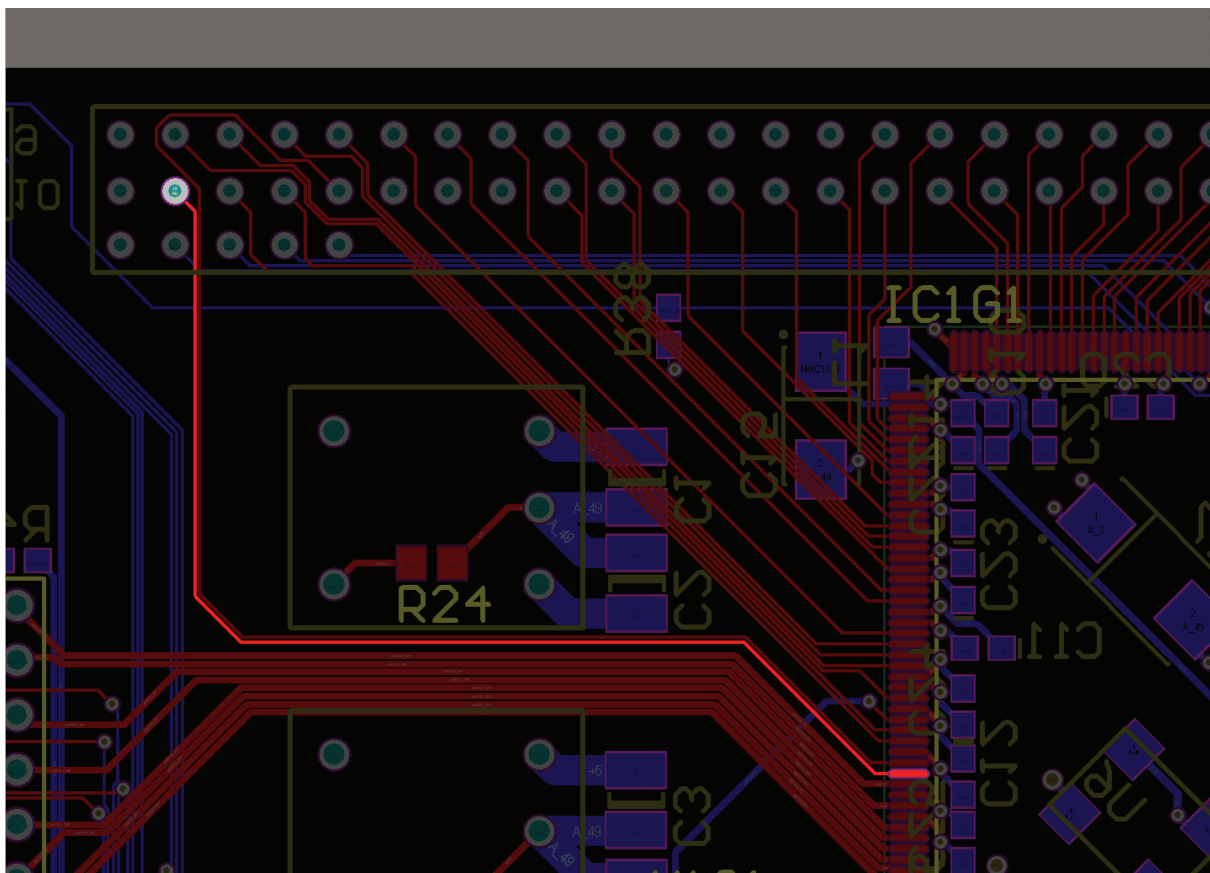
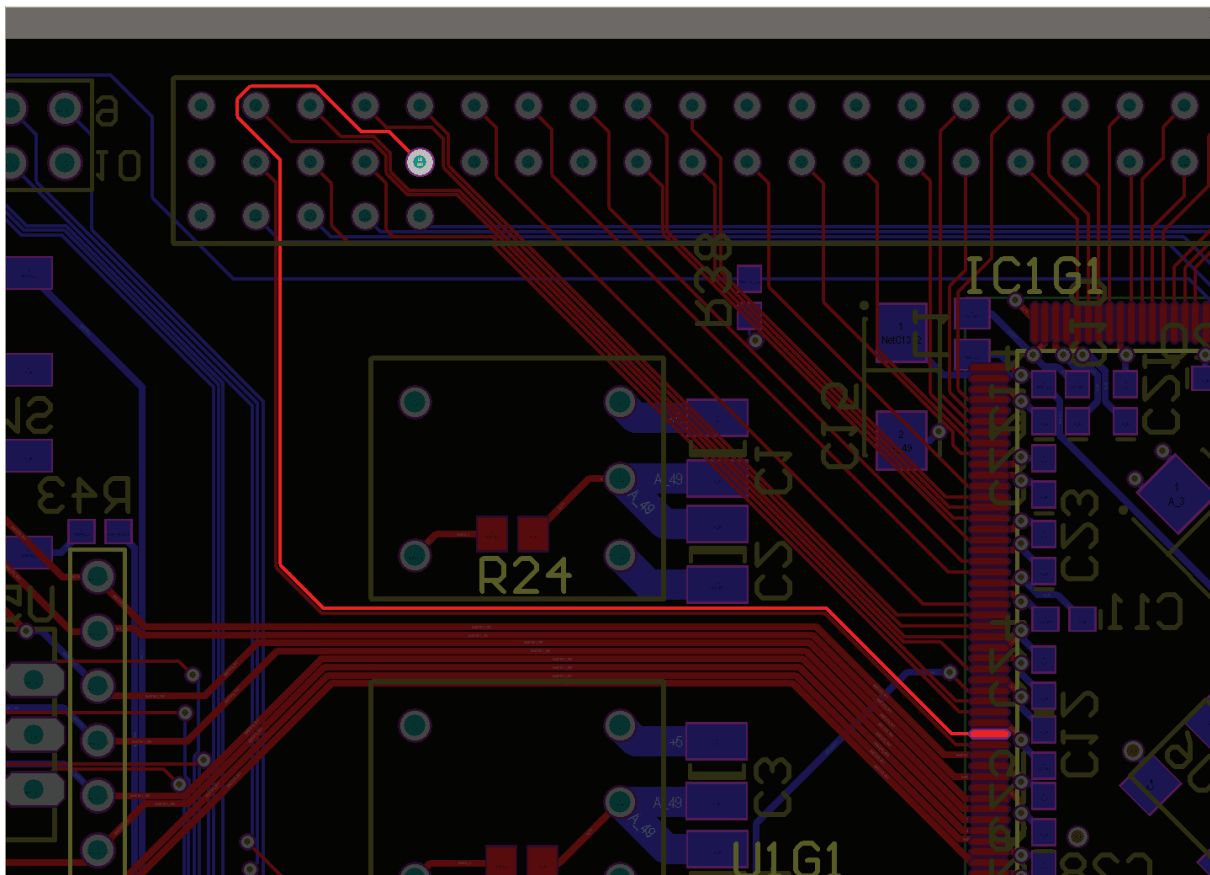
Die gewöhnliche Netzbreite wie sie in den Rules eingestellt wurde, beträgt 0,3 mm. Leider lassen sich mit einer Breite von 0,3 mm die Umleitungen auf GND nicht umsetzen. Entwe-

der die Leitungen sind zu dick oder der Abstand zwischen den zusätzlichen Vias und den ursprünglichen Vias wäre zu gering.

Die Abstandsvergrößerung zwischen den Vias ist nicht besonders günstig, da die Komponenten den verfügbaren Platz begrenzen. Das bedeutet für diese Variante schließlich, dass nur die Netzbereit verringert werden kann. Erst mit einer Netzbreite von 0,15 mm können die Leitungen gebogen werden. Da die betroffenen Leitungen für digitale Signale bestimmt sind, müssen sie keine großen Ströme führen und können daher schmaler ausgeführt werden. Die Leitungsverdünnung hat daher keine Auswirkung auf das Endergebnis. Dadurch konnte das Problem beseitigt werden.

## **7.5 Einarbeiten von Änderungen**

Die im Kapitel 4 bereits kurz vorgestellte Speicherschnittstelle am EMIF wurde in unserer Arbeit asynchron festgelegt, da die Verwendung einer synchronen Speicherschnittstelle, wie zum Beispiel SDRAM bzw. SBSRAM, für das Kommunikationssystem unseres Prototyps nicht notwendig ist. Das System würde dadurch nur komplizierter und es würden höhere Anforderungen an den gesamten Entwurf gestellt. Mit dieser Festlegung hat Michael Müller in seiner Diplomarbeit [Mül08] zwei Steuerleitungen: EMIF Steuersignale /AWE (Write enable) und /ARE (Read enable) zu FPGA auf dem Basisboard von Hand ergänzt. Als provisorische Lösung wurden zwei Kabelbrücken auf dem Bottom Layer des Basisboards von Hand gelötet. Erst nach Anschluss der FPGA PIN 191 (/ARE), PIN 192 (/AWE), und U2(/ARE), U5 (/AWE) an den Systembus im Basisboard Schaltplan wurden die entsprechende Leiterzüge im Platinen-Entwurf erzeugt. Damit wurden die beiden nachgelöteten Leitungen während des Routingprozesses eingearbeitet, wie in Abbildung 7-8 dargestellt.



**Abbildung 7-8: EMIF Steuersignal /AWE (oben), /ARE (unten)**

## **7.6 Schlusswort für das PCB Layout**

Nach dem Routen aller Leitungen wird für die GND- und VCC-Layer ein Polygon-Gitter gebaut, um alle GND-Punkte und VCC-Punkte zu verbinden. Zum Schluss wird nochmals das Design Rules Checking durchgeführt. Nach der erneuten Korrektur aller Fehler ist die PCB Konstruktion fertiggestellt.

Genaue wie bei der Top-Level-Schematic befinden sich auch hier weitere Grafiken und Tabellen im PDF Datenformat im entsprechenden Projektverzeichnis und im Verzeichnis Anhang auf der Begleit-DVD. Dort enthalten sind die komplett gefertigte Backplane sowie das Basisboard mit entsprechenden Schaltplänen und Materiallisten. Das PDF-Dokument wurde auch hier wieder im Extra-Großformat als Anhang ausgedruckt.

## 8 Das Signalintegritätsexperiment

Die Anzahl und Vielfalt digitaler High-Speed-Geräte wird heutzutage immer größer. Die Entwicklung von Boards mit schnellen Signalanstiegszeiten bei gleichzeitigem langen Routing kann zu Signalreflexionen führen. Der Designer muss sicherstellen, dass sein Leitplatten-Entwurf korrekt ist. Mit der Signalintegritätsanalyse von Altium Designer kann die Signalreflexionen und das Übersprechen bzw. die Störungen durch benachbarte Leitungen nachgebessert werden. Signalreflexionen sowie das Übersprechen werden im Kapitel 2 unter den Abschnitten 2.6.3 und 2.6.4 erörtert. Dadurch kann der Designer feststellen, ob sein Entwurf wie erwartet funktioniert. Die von Altium Designer angebotene Signalintegritätsanalyse kann sowohl unter der Schaltplan- als auch unter der Layout Phase durchgeführt werden. Bei der Analysesimulation eines Layouts ist große Aufmerksamkeit erforderlich. Der Schaltplan und das PCB Layout müssen zu einem Projekt gehören. Wenn sich unter dem Schaltplan ein zugehöriges Layout befindet, erstellt Altium Designer die Simulation immer im PCB Layout Dokument. Ansonsten, wenn Altium Designer unter dem Schaltplan kein zugehöriges PCB Dokument findet, erzeugt das Designer Programm im Hintergrund ein Standard PCB Layout, worauf die spätere Analysesimulation basiert.

Das Leiterplatten Design Werkzeug Altium bietet auch die Möglichkeit, die Reflexion und das Signalübersprechen auf den betrachteten Leitungen überprüfen zu lassen. Das Prinzip einer Signalreflexionsanalyse sowie Analyse des Übersprechens in Altium Designer wurde bereits in Kapitel 2 unter Abschnitt 2.5.6 vorgestellt. Der letzte Teil dieses Kapitels beschäftigt sich mit mehreren Experimenten, die sich auf unser Basisboard Projekt beziehen. Auf dem Basisboard wurde ein beliebiges differentiell LVDS Leitungspaar herausgesucht, auf dem die Analyse basierte. In unsere Experimenten wurden drei Schwerpunkte im Auge behalten, nämlich 1. die Reflexionsanalyse und Analyse des Übersprechens auf mit einem Loop-back System umgebauten Basisboard, 2. die Betrachtung der zeitlichen Verzögerung durch Verlängerung des LVDS Verbindung, und 3. die Beobachtung der Impulsänderung durch verdünnte Leitungen und die Distanzverbreiterung des differentiellen LVDS Verbindungspaares.

Um optimale Analyseergebnisse zu erreichen muss man zunächst die LVDS Leitungen und den I/O Standard des FPGA besser kennenlernen. Kapitel 4 vermittelt hierzu unter Abschnitt 4.1 einige Vorkenntnisse zur LVDS Technik. Die folgenden Abschnitte erklären die praktische Anwendung dieser Technik für unser Projekt und die Zusammenarbeit mit dem FPGA.

## 8.1 LVDS Standard in Cyclone FPGA

Die Cyclone II FPGAs bieten eine Multi-Protokoll-Schnittstelle, welche die Kommunikation zwischen einer Vielzahl von I/O Standards, einschließlich LVDS, LVPECL<sup>48</sup>, RSDS<sup>49</sup>, mini-LVDS<sup>50</sup>, differentielltem HSTL<sup>51</sup>, und differentielltem SSTL<sup>52</sup> ermöglicht. Durch die FPGA seitigen I/O Pins und internen Logik kann die Signal in Hochgeschwindigkeit gesendet und empfangen werden. Die Cyclone II FPGAs verfolgen den ANSI/TIA/EIA-644 Standard<sup>53</sup>, und seine LVDS I/O ist ein Hochgeschwindigkeitsschnittstelle, benutzt geringe Spannungspegel (Low voltage), Low Power, und ist auch eine General Purpose I/O Schnittstelle. In diesem Basisboard Projekt sind die LVDS Leitungen auf der FPGA I/O Bank 4 angeschlossen. Der LVDS-Empfänger unterstützt eine Datenrate von bis zu 805 Mbit/s. Gleichzeitig erreicht der Sender eine Datenrate bis zu 640 Mbit/s. Die maximale interne Taktfrequenz für einen Empfänger und Sender beträgt 402.5 MHz. Der LVDS-Standard erfordert keinen Eingangs-Referenzspannung, wird aber mit einem 100 Ohm Abschlusswiderstand zwischen den beiden Signalen am den Inputpuffer ausgerüstet.

---

<sup>48</sup> Low-Voltage Positive Emitter-Coupled Logic, es handelt es sich um eine auf Leitung optimierte Version von der PECL-Technologie

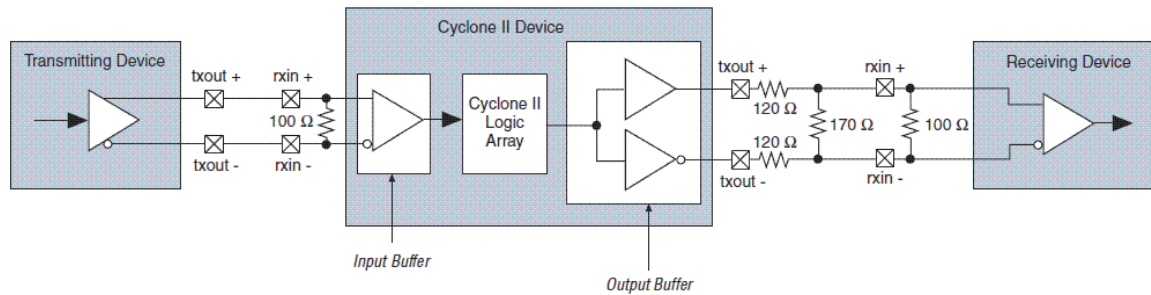
<sup>49</sup> Reduced Swing Differential Signaling

<sup>50</sup> Die Mini-LVDS ist eine unidirektionale Schnittstelle zwischen dem zeitliche Kontroller und Kolumne Treiber. Die Mini-LVDS bietet eine niedrige EMI, High-Bandwidth-Schnittstelle zum Display-Treiber.

<sup>51</sup> High-Speed Transceiver Logic

<sup>52</sup> Stub Series Terminated Logic

<sup>53</sup> Die ANSI/TIA/EIA-644-A (veröffentlicht 2001) definiert LVDS-Standard. Dieser Standard empfiehlt eine maximale Datenrate von 655 Mbit/s über Twisted-Pair-Kupferkabel, und prognostiziert einer möglichen Geschwindigkeit von über 1,9 Gbit/s für ein ideales Übertragungsmedium.



**Abbildung 8-1: Einfachen LVDS Anwendung (Darstellung aus [Alt08b])**

Die Abbildung 8-1 beschreibt die physikalische Konstruktion einer einfachen LVDS Anwendung. Eine noch detailliertere Darstellung der Funktionsweise dieser LVDS Punkt-zu-Punkt Verbindung befindet sich auch in der Diplomarbeit von C.Suffa [Suf08]. Seine Erfahrungen über LVDS tragen zu diesen Experimente wesentliches Wissen bei.

Wie im Datenblatt des Cyclone II FPGAs von Hersteller Altera [Alt08b] erläutert, liegt die Arbeitsspannung der LVDS Leitungen zwischen 1.15 V bis 1.25 V, typischerweise bei 1.2 V, während die I/O Versorgungsspannung in Bereich 2.375 V bis 2.625 V, typischerweise bei 2.5 V liegt. Die maximale Differenz zwischen den beiden Logikpegeln von High zu Low bzw. Low zu High soll unter 600 mV liegen. Der 100 Ohm Abschlusswiderstand begrenzt die Abweichung der Spannungsdifferenz auf 50 mV. Einige Werte dieses Hersteller-Datenblattes wurden zum Schluss mit dem gemessenen Ergebnissen unseres Experimentes verglichen, um nachzuprüfen, ob das FPGA auch in unserem Projekt das erwünscht Verhalten erzielt.

## 8.2 Vorbereitungen vor der Signalintegritätsanalyse

Eine gründliche Vorbereitung ist Voraussetzung einer genauen und sinnvollen Analysesimulation. Um eine erfolgreiche Signal-Integritäts-Analyse für das Layout zu gewährleisten und präzise Ergebnisse zu erhalten, müssen die folgenden Vorbereitungen vor der Ausführung der Analyse durchgeführt werden.

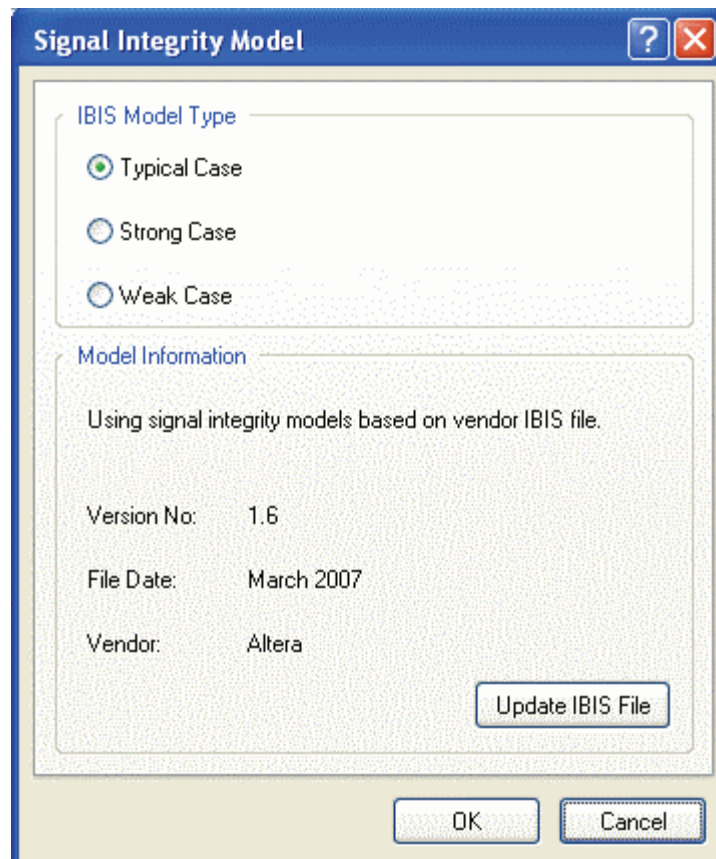
### 8.2.1 Manuelles Hinzufügen des Signal-Integrität-Modell (IBIS Modell)

Die Grundbegriffe zum IBIS Modell wurden bereits in Kapitel 2 unter dem Abschnitt 2.6.1 erklärt. Hier geht es um die Änderung und das Importieren eines IBIS Modells in Altium Designer.

Altium Designer kann nur Netze analysieren, die eine Komponente mit einem Output Pin enthalten. Das heißt, dass Komponenten wie Widerstände, Kondensatoren und Induktoren, die keinen treibenden Output Pin haben, nicht simuliert werden können.

Für jede Komponente im Entwurf muss der **Signal-Integritäts-Modelltyp** korrekt bestimmt werden. In unserem Entwurf stellen leider nicht alle Komponenten ein Signal-Integritäts-Modell zu Verfügung. Insbesondere haben die Komponenten, die von uns selbst gebaut wurden, zu solchen Problemen geführt. So können wir z.B. für den PCI 4X Express Edge Connector, die Bus Schnittstelle zu DSP Modellen und den Crystal Oscillator CFPS-73 kein passendes Signal-Integritäts-Modell feststellen. Am Fehlen der Signal-Integritäts-Modelle lässt sich in unseren Experimenten nichts ändern, weil das von uns betrachtete Netz aus LVDS Leitungen besteht, die von FPGA Output Pins zu Input Pins bzw. zum PCI Edge Connector führen. Deswegen spielt das Importieren eines passenden IBIS Modelle für den FPGA hier die entscheidende Rolle. Im Dialog Komponenten Eigenschaften auf Schaltungsplanebene kann ein Signal-Integritäts-Modell für eine Komponente hinzugefügt werden. Erst nach Hinzufügen des Modells lässt sich darunter ein IBIS Modell importieren. Die folgende Abbildung 8-2 demonstriert genau das Programmfenster, über welches man das IBIS Modelle importieren kann.





**Abbildung 8-2: IBIS Modell Importieren**

Die importierte IBIS Datei wurde von Altium Designer selbstständig bearbeitet. Die in der Datei bereits enthaltenen Pin-Definitionen werden zum Zuordnen der Pins zu den Komponenten benötigt. Für die Cyclone II EP2C8-Q208C7N FPGA, die unser Entwurf verwendet, bietet die Altera Corporation auf ihrer Website eine Download der IBIS Datei kostenfrei an. Beim ersten Importieren der heruntergeladenen IBIS Datei tauchte ein merkwürdiger Fehler auf. Das Programm meldete eine Zugriffsverletzung bei Adresse 10014B08 im Modul „colibrilib.dll“. Dabei wurde vermutlich festgestellt, dass die Pin-Zuweisung in der IBIS Datei nicht zu unserem Projekt passt. Die Altera Corporation hat für alle Cyclone II FPGA Serien eine allgemeine IBIS Datei erstellt. Der Benutzer muss diese wahrscheinlich selbst nach eigenem Bedarf verändern. Für unsere Experimente wurde das differentielle LVDS Leitungspaar NetIC1G1\_92 und NetIC1G1\_90, die beide mit FPGA Outputs Pin 92 und 90 verbindet und später durch die Leitungen A\_40, A\_39 zu Inputs Pin 88 und 87 leitet, als Testkandidaten gewählt - genau wie die benachbarte LVDS Leitung NetIC1G1\_86, die zunächst für die Analyse des Übersprechens gebraucht wird. In der IBIS Datei wurde folgende Änderungen vorgenommen.

Die Pins von 79 bis 90 in der Original IBIS Datei wurden alle als 3.3 V LVTTTL<sup>54</sup> Leitungen definiert.

	ttl133	
79	2c_ttl133_rio_d4	2c_ttl133_rio_d4
80	2c_ttl133_rio_d8	2c_ttl133_rio_d8
81	2c_ttl133_rio_d12	2c_ttl133_rio_d12
82	2c_ttl133_rio_d16	2c_ttl133_rio_d16
83	2c_ttl133_rio_d20	2c_ttl133_rio_d20
84	2c_ttl133_rio_d24	2c_ttl133_rio_d24
85	2c_ttl133_cio_d4	2c_ttl133_cio_d4
86	2c_ttl133_cio_d8	2c_ttl133_cio_d8
87	2c_ttl133_cio_d12	2c_ttl133_cio_d12
88	2c_ttl133_cio_d16	2c_ttl133_cio_d16
89	2c_ttl133_cio_d20	2c_ttl133_cio_d20
90	2c_ttl133_cio_d24	2c_ttl133_cio_d24

Die Pins 86 und 90 mussten neu als 2.5 V LVDS Leitungen definiert werden, und der in der Original Datei nicht definierte Pin 92 musste hinzugefügt werden. Die Leitungen NetIC1G1\_92 und NetIC1G1\_90, die zum differentiellen LVDS Leitungspaar gehören, mussten eine zu High und eine zu Low bzw. positiv und negativ definiert werden. Nach der Änderung sieht der betreffende Teil der IBIS Datei wie folgt aus.

	LVDS	
79	2c_ttl133_rio_d4	2c_ttl133_rio_d4
80	2c_ttl133_rio_d8	2c_ttl133_rio_d8
81	2c_lvds25_cinp	2c_lvds25_cin
82	2c_lvds25_cinn	2c_lvds25_cin
83	2c_ttl133_rio_d20	2c_ttl133_rio_d20
84	2c_lvds25_cop	2c_lvds25_co
85	2c_ttl133_cio_d4	2c_ttl133_cio_d4
86	2c_lvds25_con	2c_lvds25_co
87	2c_lvds25_cinp	2c_lvds25_cin
88	2c_lvds25_cinn	2c_lvds25_cin
89	2c_ttl133_cio_d20	2c_ttl133_cio_d20
90	2c_lvds25_cop	2c_lvds25_co
92	2c_lvds25_con	2c_lvds25_co

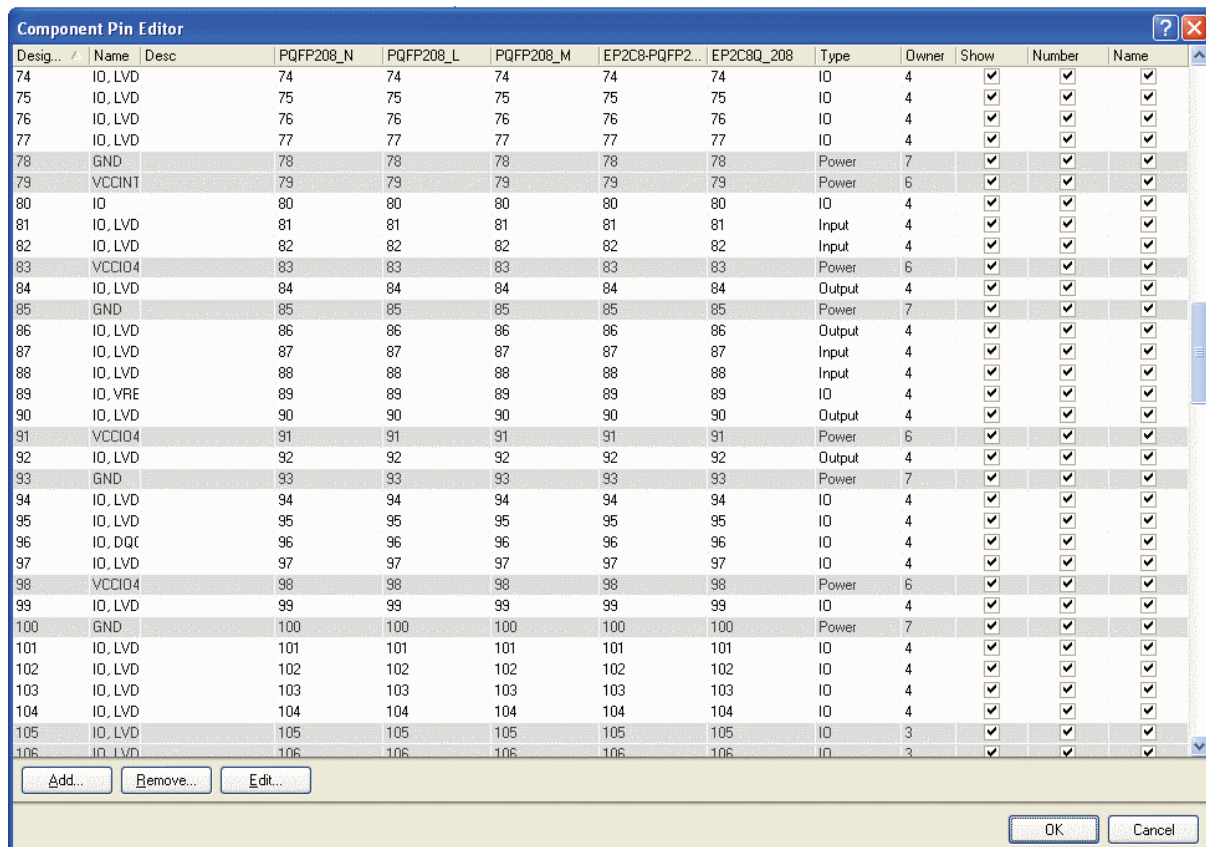
Erst nach einem erneuten Import der geänderten IBIS Datei tauchte die vorherige Fehlermeldung nicht mehr auf. Die originalen und die geänderten IBIS Modelle Dateien stehen im Verzeichnis IBIS Modelle auf der Begleit-DVD zur Verfügung.

## 8.2.2 Bearbeiten der FPGA Schematic Pins

Altera Cyclone II FPGA besitzt über 208 Pins. Die Pins verleihen der FPGA ihrer elektrischen Eigenschaften und definieren die Verbindungspunkte auf der FPGA für die Leitung der ein- und ausgehenden Signale. Entweder durch die Aktivierung des „Komponente Pin Editor“

<sup>54</sup> Low Voltage Transistor Transistor Logic

im Dialog Komponente Eigenschaften oder später per **Edit Buffer** Option im Signal Integrität Panel (siehe Abschnitt 8.2.5) können die Pins des FPGAs konfiguriert werden. Abbildung 8-3 verdeutlicht die Konfiguration der FPGA Pins mit dem Komponenten Pin Editor.

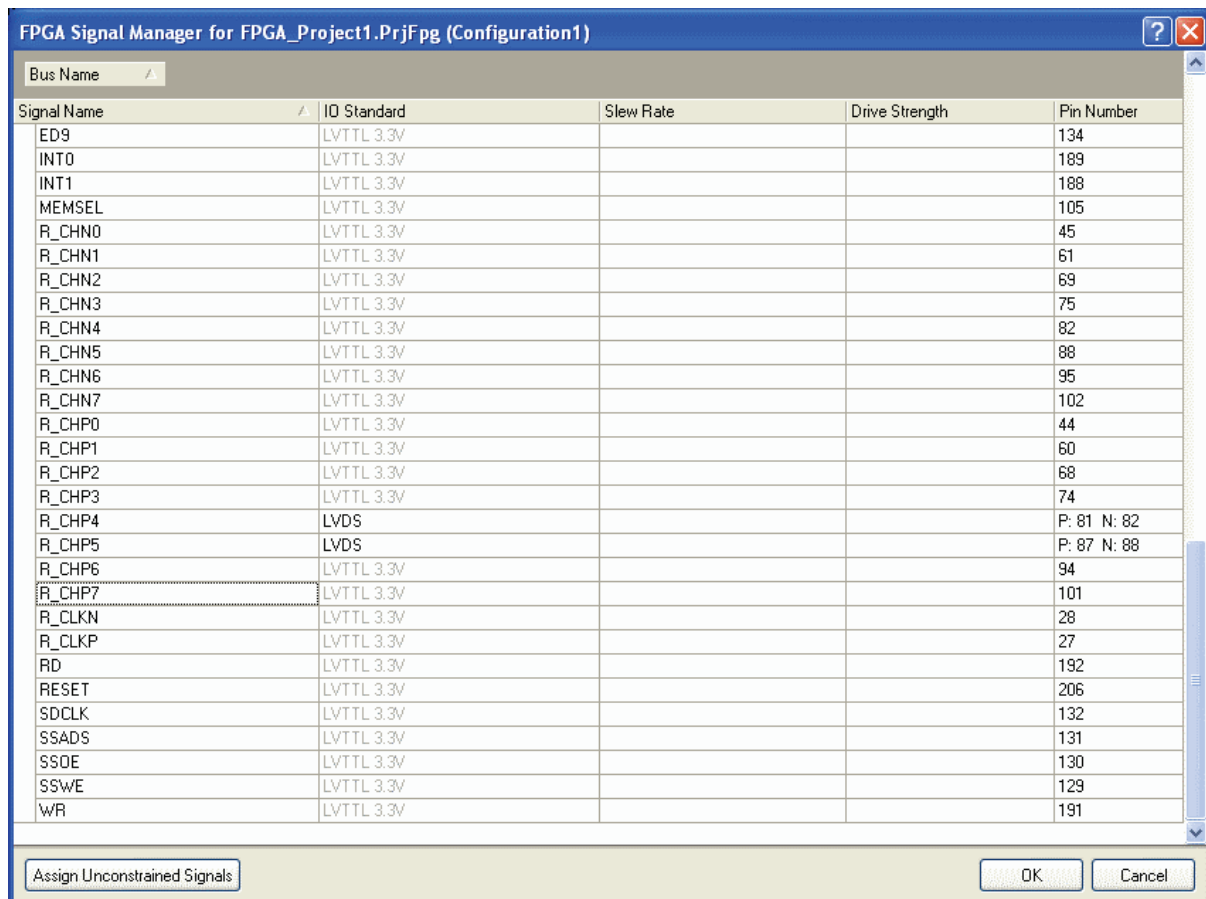


**Abbildung 8-3: Komponente Pin Editor**

Die Standardeinstellung für die Pins sind normale bidirektionale I/O Pins. Für unsere Experimente müssen die Pins jedoch streng nach Eingang oder Ausgang neu definiert werden. Darum werden die Pins 92 und 90 bei FPGA als Output (Ausgang), die Pins 88 und 87 als Input (Eingang) konfiguriert. Dasselbe Prinzip gilt auch für die Pins 86, 84 bzw. die Pins 82, 81.

### 8.2.3 FPGA Signal Manager

Die Signal Integrität von Altium Designer kann verwendet werden, um die optimalen Einstellungen für einen bestimmten FPGA Pin herauszufinden. Zum Beispiel sind I/O Pins in einem FPGA Entwurf für Datenleitungen geeignet und müssen auch so definiert werden. Die elektrischen Eigenschaften für die Pins des physischen FPGA Gerätes können im FPGA Signal Manager eingegeben werden. Abbildung 8-4 zeigt den FPGA Signal Manager Dialog, wobei jeder Pin und seine elektrischen Eigenschaften detailliert dargestellt werden.



**Abbildung 8-4: FPGA Signal Manager**

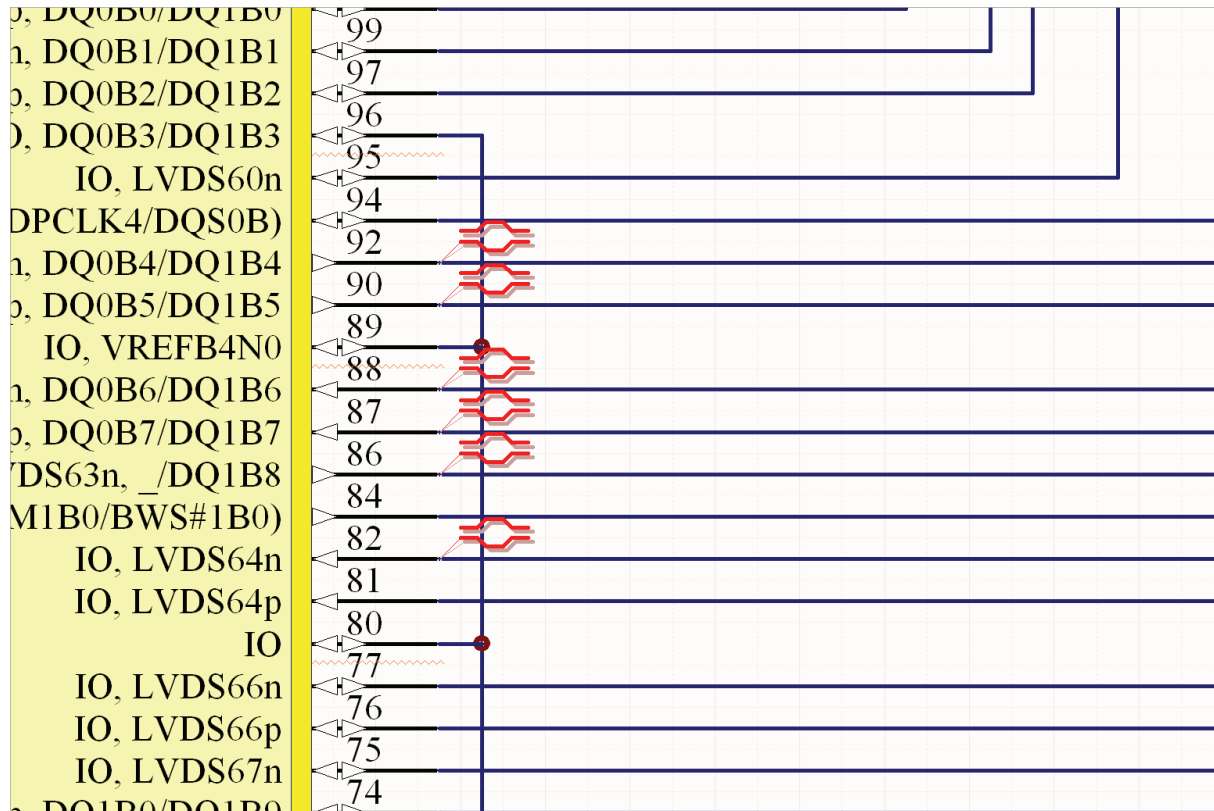
Hier können für jeden Pin der I/O Standard, die Anstiegsrate, die Abfallrate und deren Stärke eingestellt werden. Die verfügbaren Anstiegs- oder Abfallgeschwindigkeiten und Stärken hängen von der aktuellen Einstellung der I/O Standards der Pins ab. Die definierten Signal-Informationen werden in der entsprechenden Constraint Datei (Constraint 1.Constraint) für das FPGA Projekt gespeichert. Für unser Experiment lag kein extra FPGA Projekt vor. Darum wurde eine Standard FPGA Projekt für das Altera Cyclone II FPGA bei der ersten Einrichtung eingefügt. Erst danach kann der I/O Standard der Pins 92, 90 und der Pins 86, 84 auf der Seite des Senders, sowie der I/O Standard der Pins 87, 88 und der Pins 82, 81 auf der Seite des Empfängers als LVDS festgestellt werden.

Die Constraint Dateien und das FPGA Projekt sind spezifische Alitum Designer Dateiformate. Sie wurden für jedes Experiment einmal erstellt und befinden sich in den entsprechenden Verzeichnissen auf der beiliegenden DVD.

## 8.2.4 Signal Integrität Design Regeln Einstellung

Für genaue Signal-Integritäts-Analyseergebnisse müssen noch spezifische PCB Entwurfs Regeln, wie z.B. die passenden Signal Stimulus und Spannungswerte für die GND und VCC im

Schaltplan als Parameter oder später im PCB Layout hinzugefügt werden. Durch die Platzierung der Anweisung zu den ausgewählten LVDS Leitungen auf der Schaltplanebene wird ersichtlich, dass sie zum Differentiellen Paar gehören. Die Entwurfs Regeln können den dort platzierten Anweisungen angehängt werden. In Abbildung 8-5 werden die Differentiellen Leitungspaare durch rote Anweisungssymbole definiert.



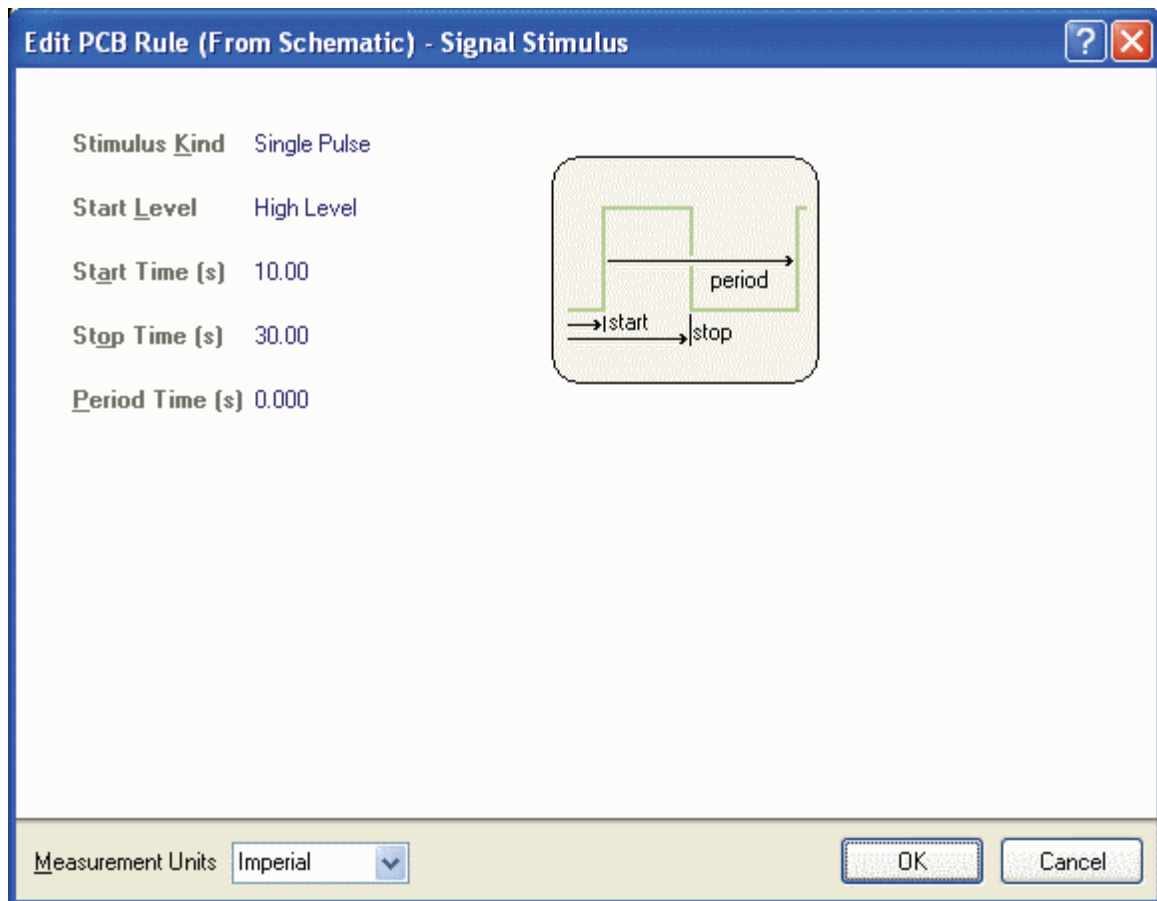
**Abbildung 8-5: Anweisungen für Differentielle Paare**

Der Signal Stimulus vergibt die Eigenschaft des Signal-Reizes bei Durchführung einer Signal Integrität auf einen Entwurf und beschreibt dabei das eingespeiste Signal an jedem dem zu prüfenden Netz angeschlossenen Output Pin. Die Einstellung für den Stimulus umfasst folgende Punkte: [Alt08a]

- Bei einigen **Stimulus Typen** kann der Signal Stimulus entweder als eine konstante Spannung (Constant Level), als ein einzelner Impuls (Single Pulse) oder als eine kontinuierliche Impulsfolge (Periodic Pulse) definiert werden.
- Das **Start Level** gibt die konstante Spannung für das Constant Level oder das erste Spannungsniveau für den Puls-basierten Stimulus an.

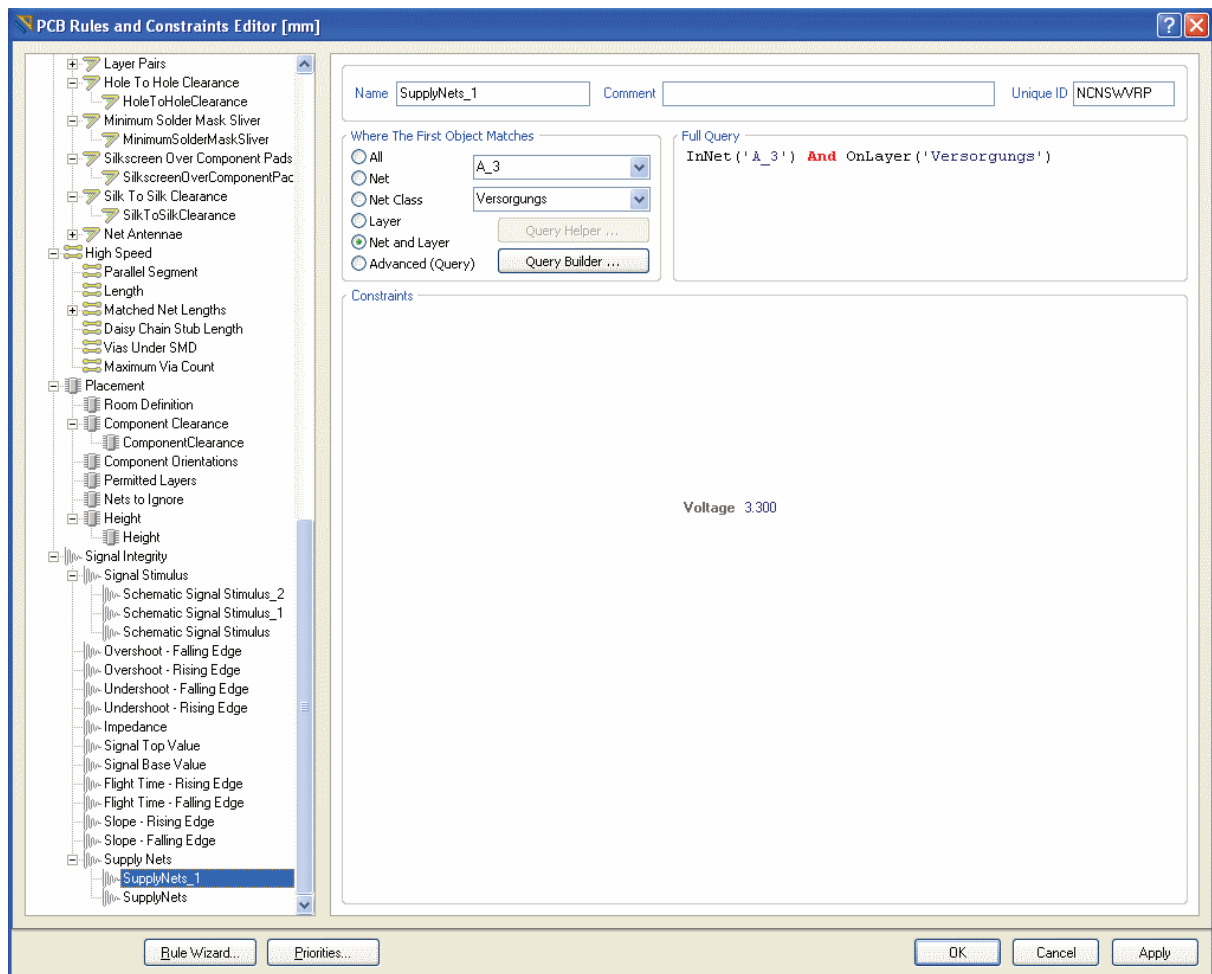
- **Start Time** und **Stop Time** sind die Start- und die Stoppzeit für den Puls basierte Signal Stimulus. Sie werden zur Berechnung der Breite des Impulses benötigt.
- **Period Time** ist die Zeit zwischen den Impulsen einer periodischen Impulsfolge. Nachdem die erste Zeitperiode abgelaufen ist folgt ein weiterer, identischer, wieder mit Startzeit und Stoppzeit eingespeister Impuls.

In unserem Experiment wurden den Anweisungen auf dem Differentiellen Leitungspaar NetlC1G1\_92, NetlC1G1\_90 bei LVDS (Sender), sowie NetlC1G1\_86, NetlC1G1\_84 (Empfänger) die entsprechenden Stimulus Regeln angehängt. Wie aus Abbildung 8-6 hervorgeht wurden alle Stimulus Typen als Single Pulse eingestellt. Mit der Start Level Option wurden die den negativen Pins 92 und 86 angeschlossenen Leitungen NetlC1G1\_92 und NetlC1G1\_86 auf High Level, und in Gegensatz dazu die Leitungen NetlC1G1\_90 und NetlC1G1\_84 auf Low Level eingestellt. Die Integritätsanalyse aller Signale begann ab 10 ns und endete um 30 ns, wobei die Pulsänderungen in nur einer Zeitperiode betrachtet wurden. Nach Aktualisierung des PCB Dokuments wurden die eingefügten schematischen Regeln automatisch auch im PCB Layout erstellt.



**Abbildung 8-6: Signal Stimulus**

Außer den Stimulus Regeln müssen auch die Spannungswerte für GND und VCC den Design Regeln hinzugefügt werden. Das Basisboard wird mit 3.3 V betrieben. Dazu können die Spannungswerte 0 V für GND, das ist das Netz A\_49 und auch der komplette GND Layer, bzw. 3.3 V für VCC, Netz A\_3 und der komplette Versorgungslayer unter **Supply Nets** im Dialog Design Rules unter **Signal Integrity** eingegeben werden. Abbildung 8-7 zeigt die Regeleinstellung für die VCC Leitungen und Layer.



**Abbildung 8-7: Regeleinstellung für Versorgungsleitungen und Layer**

(Anmerkung: Die Regeln können erst auf Schaltplanebene vordefiniert werden. Nach Aktualisierung des PCB Dokuments wurden sie automatisch in Layout Rules erzeugt. Man kann sie später auch selbst im PCB Layout unter dem Dialog Design Rules festlegen).

## 8.2.5 Bauteil Technologie

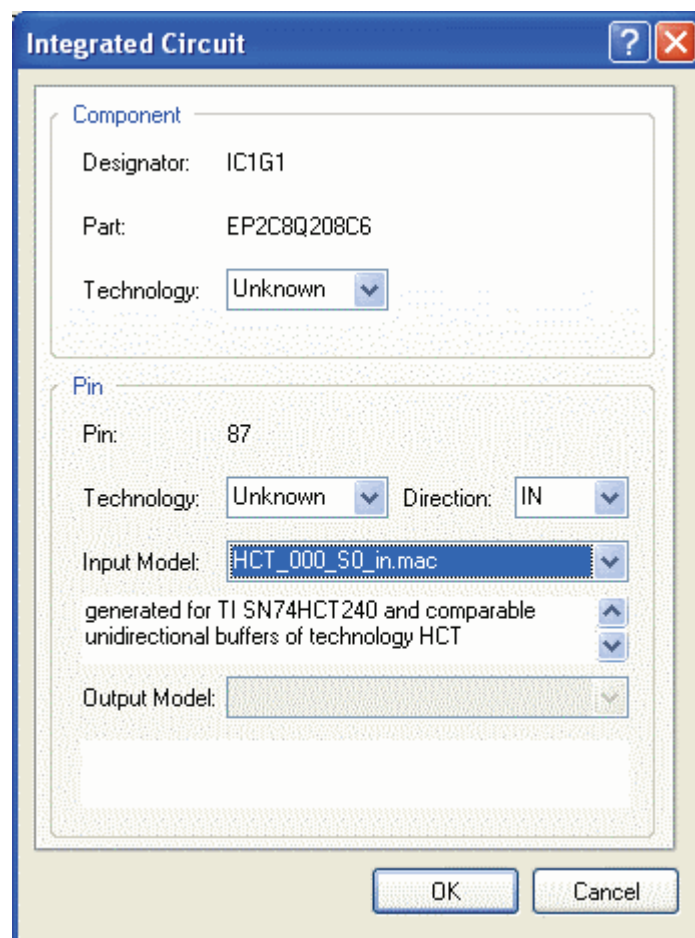
Vor der Ausführung der Signalintegritätsanalyse müssen schließlich noch die richtige Komponenten Technologie (FPGA Technologie), die Pin Eigenschaften, Input- und Output Modelle sowie die Pin Richtung konfiguriert werden.

Die FPGA-Technologie wird von Hardwareingenieuren mit Programmierwerkzeugen eingesetzt, die für Embedded-Experten konzipiert sind. Der Chiphersteller stellt neben der Herstellung des Produktes auch die Werkzeuge und Design Software zu Verfügung, welche die Ei-



enschaften von FPGAs darstellen und die Vorteile der FPGA-Technologie für neue Anwendungen zur Verfügung stellen, z.B. die IBIS Modelle und die Quartus II<sup>55</sup> Software.

Die Änderung der Komponenten Technologie erfolgt mittels **Edit Buffer** Option nach Aufruf der Signal Integrität durch Klicken der rechten Maustaste in die Liste der Pins im Signal Integrität Panel. Außerdem kann hier nochmals die Pin-Richtung bestimmt werden bzw. ob die Pins zum Eingang oder Ausgang gehören sollen. Die Wirkung ist die gleiche wie bei dem schon in Kapitel 8.2.2 vorgestellten Komponenten Pin Editor. Aus Abbildung 8-8 ist die soeben erwähnte Einstellung ersichtlich.



**Abbildung 8-8: Edit Buffer Option**

Hier hat Altium Designer schon einen Standard TI SN74HCT240 HCT Technologie mit 5 V Spannung auf dem Ausgangspin vorgeladen. Leider ist dies für unser Projekt völlig ungeeignet, weil die Output Pins auf 2.5 V LVDS eingestellt werden müssen. Die passenden Komponenten und die entsprechende Pin Technologie können erst nach dem Importieren des IBIS

<sup>55</sup> Das FPGA Entwurfs- und Modellierungswerkzeug der Altera Corporation.

Modelle ausgewählt werden. Die vom IBIS Modell stammende LVDS Technologie wird in typischer, starke und schwache Fälle unterteilt. In diesem Experiment wird nur der typische Fall im Auge behalten. Die entsprechenden Dateinamen lautet also X2c\_lvds25\_cin\_tp\_in.mac für den Input und X2c\_lvds25\_co\_tp\_out.mac für den Output.

### **8.3 Signalintegritätsanalyse mit dem Demoprojekt**

Im Verzeichnis **Examples \ Signal Integrity** befindet sich eine Reihe von Demoprojekten. Das dort enthaltene Differential Pair Projekt illustriert die Signalintegritätsanalyse einer einfachen LVDS Übertragungsleitung zwischen zwei FPGAs mit differentiellen Leitungspaar. Es ermöglicht, die Signalintegritätsanalyse klar und deutlich zu demonstrieren. Es verschafft uns einige Vorkenntnisse über den Wertbereich der Ausgangs- bzw. Eingangsspannung und veranschaulicht die Impulse. In Abbildung 8-9 wird die Konstruktion des Demoprojekts im Schaltplan und im PCB Layout erklärt. Die hier verwendeten zwei FPGAs stammen von der Firma Xilinx Model Spartan-IIe XC2S200E. Die Beiden FPGAs sind mit einem differentiellen Leitungspaar unter Verwendung von LVDS Technik miteinander verkoppelt. Im Vergleich zu unserem Projekt sind die beide Projekte im allgemeinen Sinn ähnlich gestaltet.

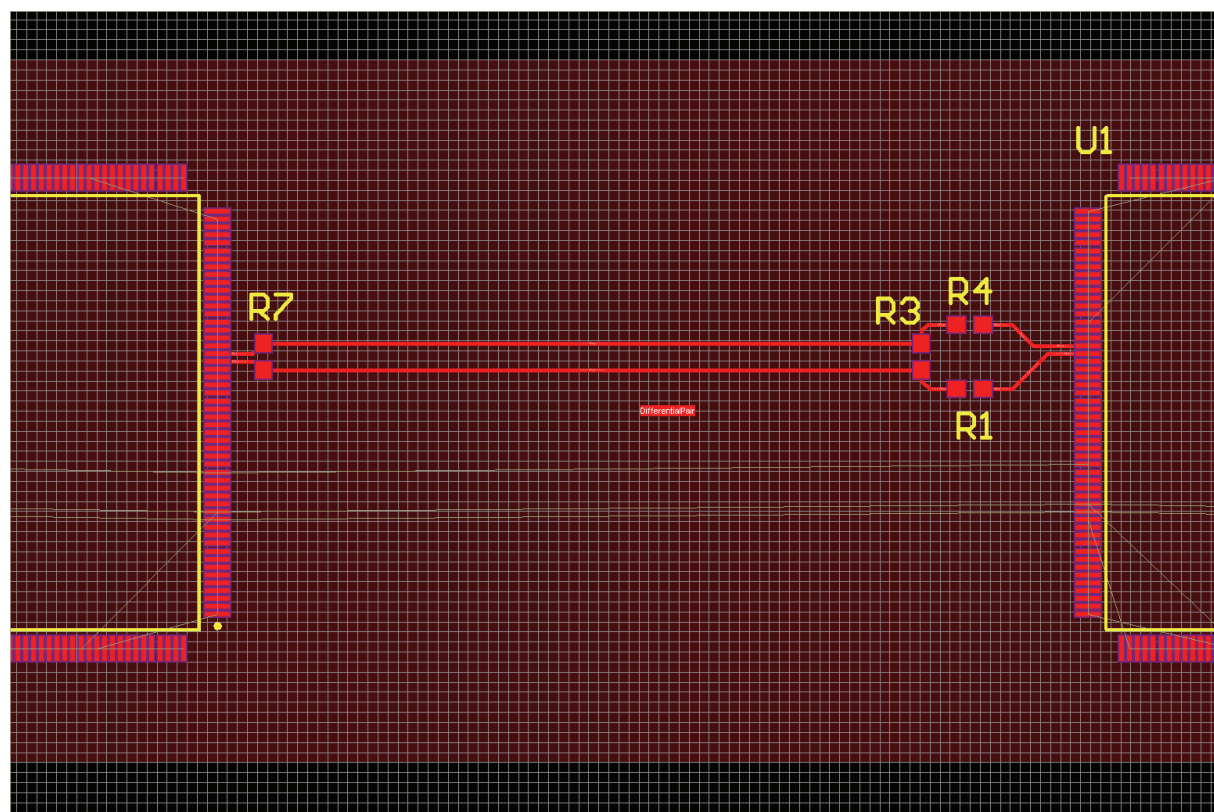
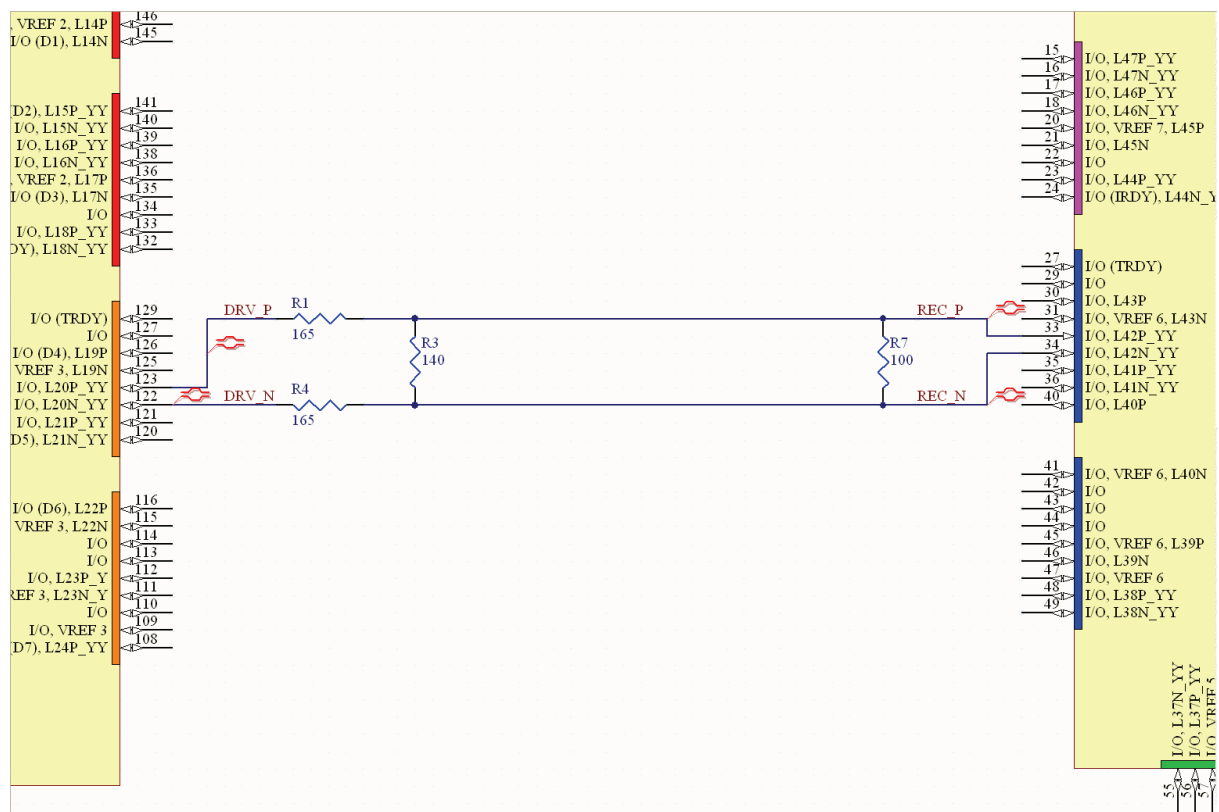


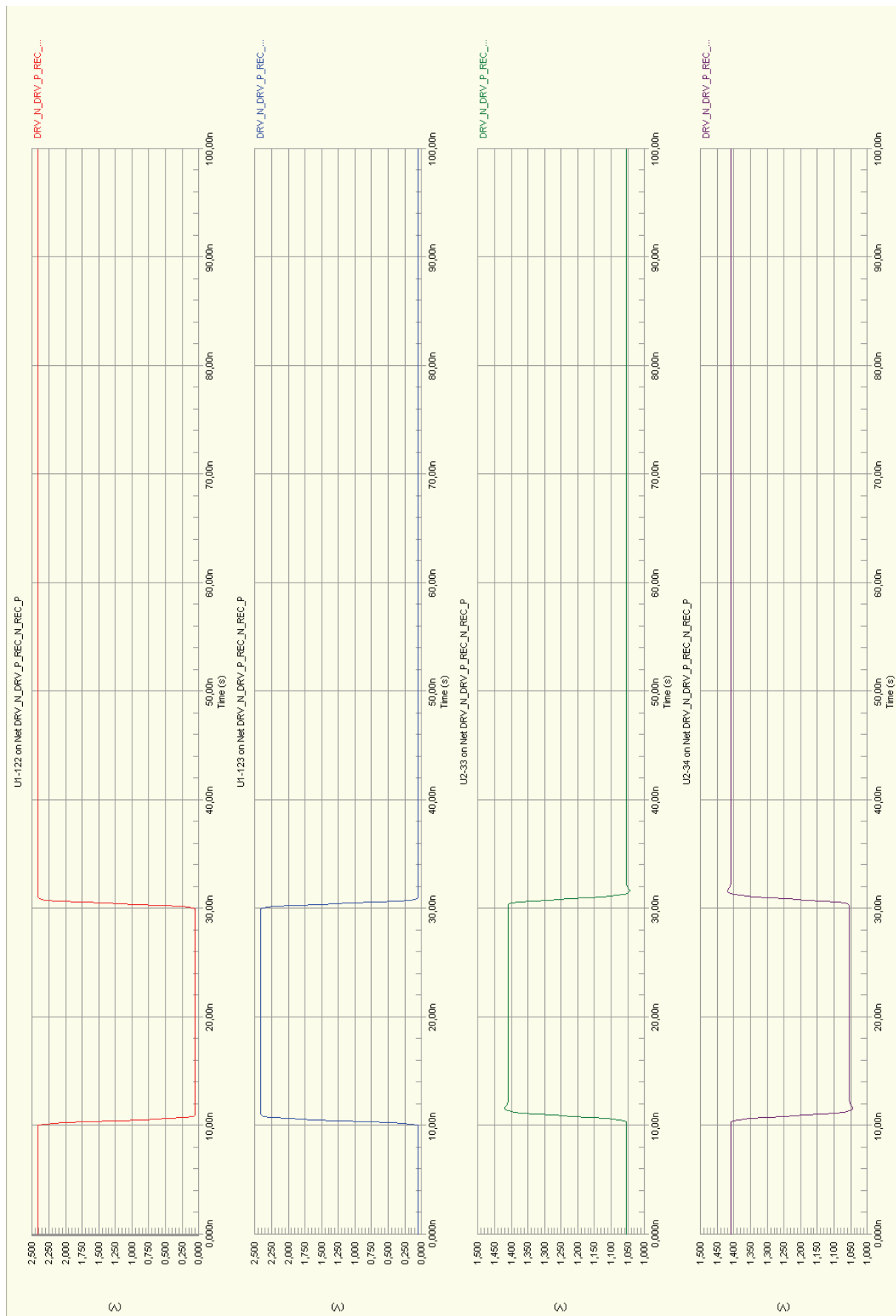
Abbildung 8-9: Das Demoprojekt in Schaltplan(oben) und Layout(unten)

## 8.4 Das Ergebnis des Demoprojekts

Mit dieser Reflexionenanalyse können wir sicherstellen, dass die Spannung auf den beiden Ausgangspins 122 und 123 bei 2.414 V liegt, wie in Abbildung 8-10 dargestellt. Dies passt zur Beschreibung über die typische Versorgungsspannung 2.5 V einer normalen LVDS Schnittstelle.

Ungefähr bei 10 ns wurde ein Signal auf Leitung DRV\_N eingespeist. Damit beginnt der Wechsel von High zu Low. Die Spannung fällt also von Top Line 2.414 V auf Basis Line 50.24 mV. Nach einer Periode von 30 ns schwenkt der Spannungspegel wieder zur Ausgangssituation zurück. Die Leitung DRV\_P wurde im Start Level in Design Rule als Low Level definiert. Darum sieht sein Impuls komplett umgekehrt aus. Während der Analyse erreicht der Impuls auf Leitung DRV\_N sein Maximum 2.415 V bei 32.18 ns und sein Minimum 50.04 mV bei 12.24 ns. Impuls Maximum 2.415 V und Impuls Minimum 50.07 mV für die Leitung DRV\_P liegen aber bei 12.14 ns und 32.28 ns. Durch die vier Widerstände auf dem LVDS Leitungspaar verringert sich die Spannungen auf den Eingangspins 33 und 34 um ungefähr 1.005 V auf 1.410 V. Durch Nachschlagen im Produktionsdatenblatt von Xilinx [Xil08] bestätigte sich, dass die durch unserer Auswertung ermittelten 1.410 V dem im Datenblatt stehenden typischen Wert von 1.425 V sehr nahe kommt. Somit lässt sich durch das Demoprojekt das I/O Verhältnis der Komponenten gut und richtig erklären. Die Signalreflexion wurde in diesem Fall sehr gut unterdrückt. Das positive Überschwängen betrug hier nur 9 mV. Die Impulsformen verhielten sich während der Analyse sehr stabil (siehe Abbildung 8-10). Wegen der Ähnlichkeit der Anwendung zur LVDS Technik sollte sich die in unserem Experiment betrachtete Impulsform auch ähnlich bewegen.

Der komplette Schaltplan und die Layout Darstellungen wurden als PDF Format exportiert und können im Verzeichnis Anhang auf der Begleit-DVD nachgeschlagen werden.



**Abbildung 8-10: Reflexionsanalyseergebnis des von Demoprojektes**

## 8.5 Benötigte Designänderungen für die Signalintegritätsanalyse

Während die Signalintegritätsanalyse läuft, werden die Spannungen auf allen mit LVDS Leitungen verbundenen Schnittstellen ausgemessen. Dabei wird jedes LVDS Leitungspaar auf dem Basisboard Layout getrennt und auf dem Top Layer bzw. Bottom Layer platziert. Die mit den Ausgangspins verbundenen Leitungen befinden sich auf dem Bottom Layer und führen nicht zu den Eingangspins, sondern direkt zu dem PCI Express 4X Steckkartenverbinder. Dieser Konstruktion dient zur Kommunikation mit dem anderen Basisboard, wenn sie alle auf der Backplane eingesteckt sind. Mehr zur Konstruktion des gesamten Kompaktsystems und zu den Grundlagen der verwendeten Hardware findet man im Kapitel 3. Zur Realisierung der Analyse des I/O Verhältnisses eines Basisboards müssen einige Designänderungen vorgenommen werden. Aus diesem Grund wurden auf dem Basisboard einige Loopback Systeme eingebaut. Das Loopback System ist nichts weiter als ein Via, das für den Verbindungsaufbau zwischen den LVDS Leitungen auf dem Bottom Layer und den LVDS Leitungen auf dem Top Layer sorgt.

### Designänderungen für das erste Experiment

Bei diesem Versuch handelt es sich um die Signalreflexionsanalyse und Analyse des Übersprechens, wobei ein bestimmtes LVDS Leitungspaar betrachtet wurde. Dazu wurden auf der Basis des originalen Basisboard zunächst zwei Loopback Systeme eingebaut und ein beliebiges LVDS Leitungspaar als Versuchskandidaten ausgewählt.

- Eine der ausgewählten LVDS Leitungen besteht aus den Netzen NetIC1G1\_92 und A\_40. Netz NetIC1G1\_92, ist auf der Seite des Senders mit dem Ausgangspin 92 des FPGAs verbunden und führt zum Loopback System. Netz A\_40 ist vom Loopback ausgehend mit dem Eingangspin A40 B38 des Steckkartenverbinders und mit dem Eingangspin 88 des FPGAs verbunden.
- Die andere Leitung des LVDS Leitungspaares besteht aus den Netzen NetIC1G1\_90 und A\_39. Netz NetIC1G1\_90 ist mit dem Ausgangspin des FPGAs 90 verbunden und führt zum Loopback System. Netz A\_39 ist vom Loopback ausgehend mit den Pins des Steckkartenverbinder A39, B37, und dem Eingangspin 87 des FPGAs verbunden.

Wie in Abschnitt 8.1 erwähnt verwendet die allgemeine LVDS Standard Technik auf der Seite des Empfängers einen Abschlusswiderstand, z.B. R18 mit einem Wert von 100 Ohm. Dieser Abschlusswiderstand ist verbunden mit dem differentiellen LVDS Leitungspaar R\_CHN5,

R\_CHP5. Bei der anschließenden Analyse des Übersprechens hat der Altium Designer die beiden Leitungspaare wie ein gesamtes Netz betrachtet. Dies führt jedoch nicht zu einem korrekten Analyseergebnis. Die Analyse des Übersprechens bezieht sich auf zwei benachbarte Leitungen. Deswegen wurde zum Zweck der Analyse des Übersprechens eine zusätzliche LVDS Leitung aus einem benachbarten differentiellen Leitungspaar als dritter Versuchskandidat ausgesucht.

- Die dritte Leitung besteht aus den Netzen NetIC1G1\_86 und A\_36. NetIC1G1\_86 ist mit dem Ausgangspin des FPGAs 86 verbunden und führt weiter zum Loopback System. Netz A\_36 ist mit den Eingangspins A36, B34 des Steckkartenverbinders und dem Eingangspin 82 des FPGAs verbunden.

(Anmerkung: Die Leitungen mit den Namen A\_40 und A\_39 auf der Seite des Senders hießen ursprünglich T\_CHN5 und T\_CHP5. Die Leitungsamen wurden nur auf Grund der Einfügung der Top-Level-Schematic in A\_40 und A\_39 geändert, Siehe dazu Kapitel 6.1 Top-Level-Schematic).

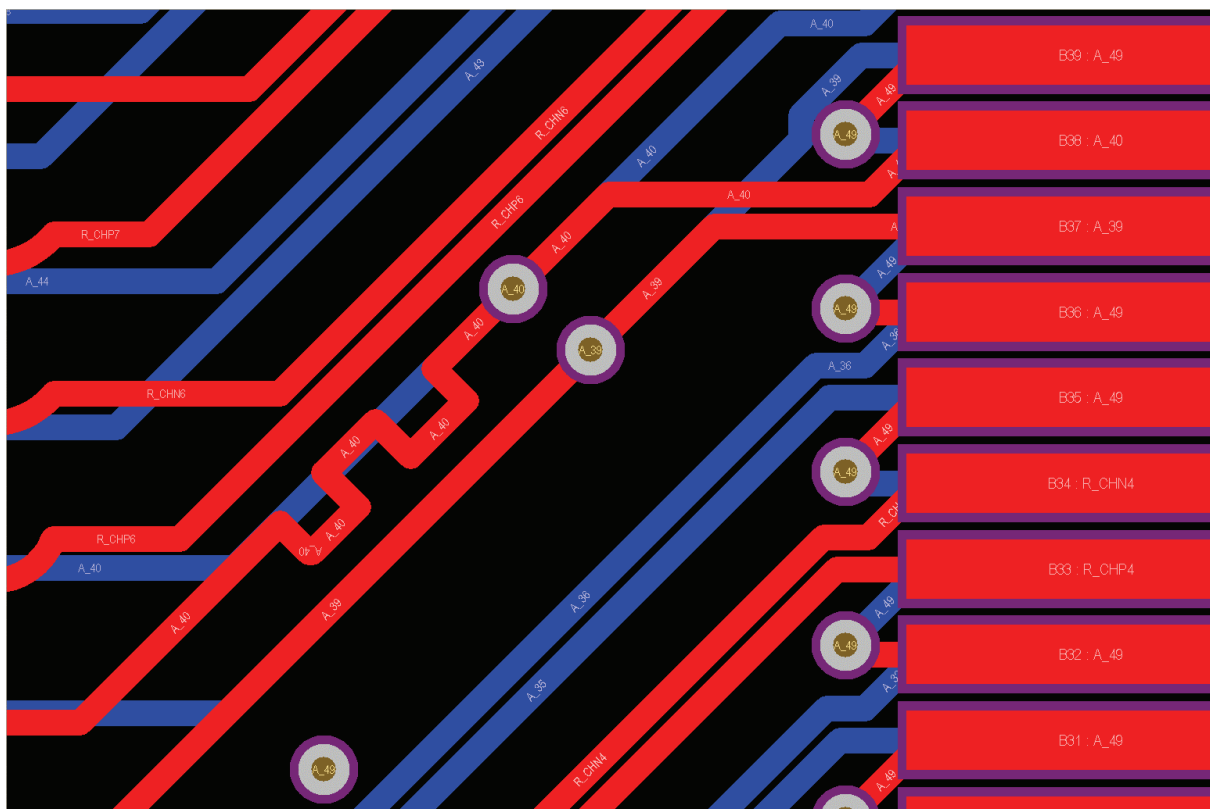
In Abbildung 8-11 ist das Loopback System sowohl auf Schaltplan Ebene als auch auf Layout Ebene zu sehen. Nach der Aktualisierung des PCB Dokuments kann das entsprechende Vias auf dem PCB Entwurf eingefügt werden. Durch den Einbau des Loopback Systems wurden dabei Teile der Leitungen R\_CHN5, R\_CHP5 und R\_CHN4 den Leitungen A\_40, A\_39 und A\_36 zugeordnet. Dadurch haben sich auch die Leitungslängen der Leitungen A\_40, A\_39 und A\_36 geändert. Weil die drei Leitungen schon zuvor mit anderen LVDS Leitungen zu einer Netzklasse zusammengefasst worden waren, wurden nun alle Leitungen durch die Interactive Length Tuning Funktion (mehr zu Interactive Length Tuning Funktion folgt im Kapitel 7 unter Abschnitt 7.4) gleichlang gemacht. Jetzt hat das Netz seine Länge verändert und passt nicht mehr zur früheren Netzklasse. Es muss daher an dieser Stelle aus der früheren Netzklasse herausgenommen und in einer neuen Klasse mit dem Namen Signal 7 erfasst werden. Nun muss nochmals die Netzlänge angepasst werden. Der PCB Entwurf sieht schließlich wie in der folgenden Abbildung aus.





## Designänderungen für das zweite Experiment

Dieser Versuch zeigt durch die gebogene LVDS Leitung die zeitliche Verzögerung auf den Eingangspins des FPGAs. Der Umbau folgt auf den vorherigen ersten Versuch. Auf das mit dem Loopback System ausgerüstete Basisboard wurde die LVDS Leitung A\_40 auf dem Top Layer zweimal hintereinander gebogen (siehe Abbildung 8-12). Die gebogene Leitung verursacht auch das Problem mit der Leitungslänge. Weil die Netze A\_40 und A\_39, die sich in derselben Netzklasse befinden, nicht mehr gleichlang sind, muss die Interactive Length Tuning Funktion nochmals zu Einsatz kommen.

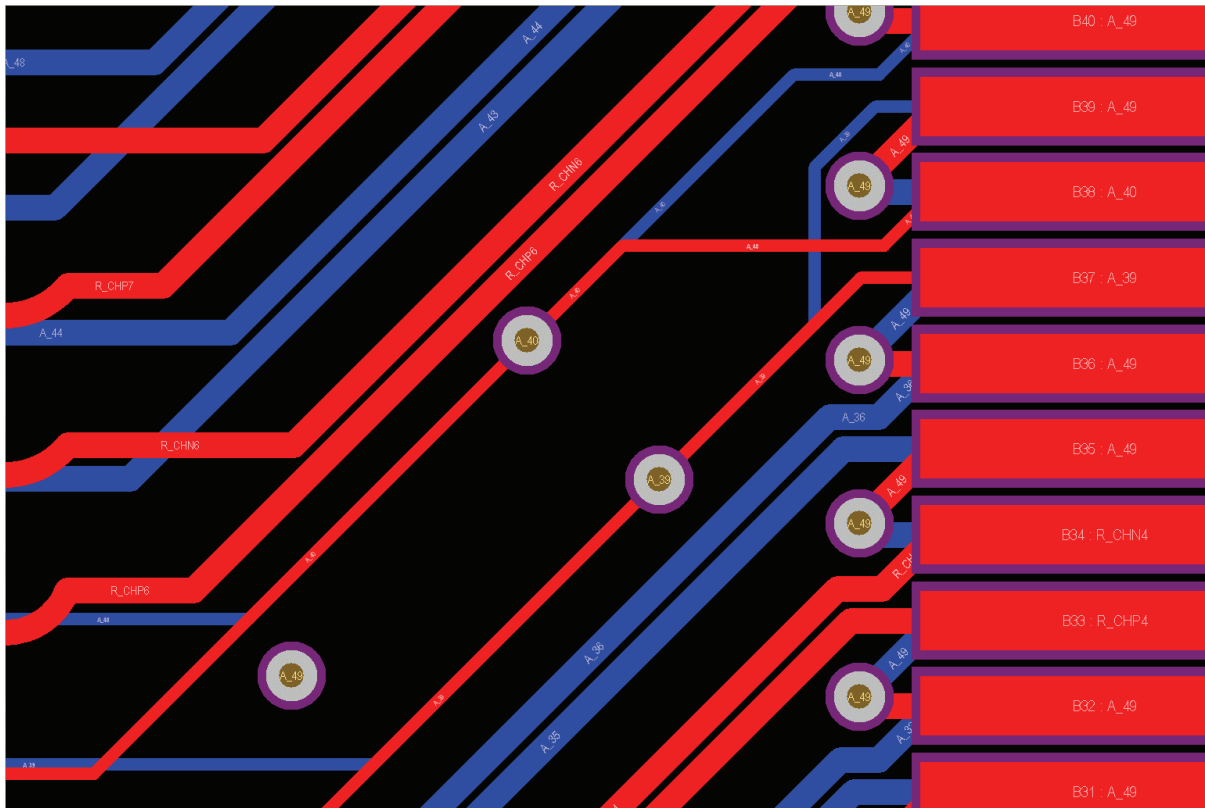


**Abbildung 8-12: gebogene LVDS Leitung**

## Designänderungen für das dritte Experiment

Dieser Versuch veranschaulicht die Spannungsänderung auf den Eingangspins des FPGAs, während sich die Impedanz durch die verringerte Leitungsbereite und den größeren Abstand zwischen zwei Loopback Systemen erhöht. Abbildung 8-13 verdeutlicht, dass auf der Basis des im ersten Versuch veränderten Entwurfs vor allem die Bereite der LVDS Leitungen A\_40 und A\_39 von normalerweise 0.3 mm auf 0.15 mm verringert wurde. Außerdem wurde versucht, den Abstand zwischen zwei Vias des Loopback Systems soweit wie möglich zu ver-

größern. Selbstverständlich musste danach auch die Länge der beiden Leitungen erneut angepasst werden.



**Abbildung 8-13: Verdünnte LVDS Leitungen und Vergrößerter Abstand zwischen zwei LVDS Leitungen**

## 8.6 Ergebnis der Experimente

Dieser Abschnitt behandelt die Ergebnisse der vorliegenden Experimente. Im Vergleich mit dem im vorherigen Abschnitt 8.3 ausgewerteten Analyseergebnis des Demoprojekts und auch mit den im Hersteller-Datenblatt [Alt08b] stehenden Werten stellte sich heraus, dass die Spannungsimpulsform der FPGA Eingangspins wie erwartet dargestellt wurde. Die Experimentergebnisse wurden in speziellen sdf-Dateien skizziert und befinden sich auf der Begleit-DVD im Projektverzeichnis jedes Experimentes. Die erzeugten sdf-Dateien können nur mit Hilfe von Altium Designer geöffnet werden.

### Das Ergebnis des ersten Experimentes

Die Konstruktion unseres Experimentes ist komplizierter als die des Demoprojektes. Während jede LVDS Leitung des Demoprojektes nur einen Ausgang und einen Eingang besitzt, hat jede LVDS Leitung auf unserem Basisboard einen Ausgang und drei Eingänge. Zwei Eingänge auf der Seite der PCI Express 4X Steckkarteverbinder wurden auf den Top Layer und den

Bottom Layer der Leitplatte verteilt. Wichtig ist, dass von Ausgangspin des FPGAs aus noch ein Weg durch das eingebaute Loopback System zum Eingang desselben FPGAs führt.

Die untere Abbildung 8-14 zeigt ein ausgezeichnetes Reflexionsanalysediagramm. Mehr zur Signalreflexion wurde bereits in Kapitel 2 unter dem Abschnitt 2.6.3 behandelt. Bei allen verbundenen Eingängen wurde mit einer herkömmlichen Standard LVDS Ausgangsspannung von 2.414 V gerechnet. Die Signalreflexion ist in der Kurvendarstellung bei alle Eingängen deutlich zu erkennen. Bei der Beobachtung z. B. des Pin 87 der Komponente IC1G1 beginnt die Signalübertragung ungefähr bei 10 ns auf der Base Line mit 999.4 mV. Dann erreichte sie in einer Anstiegszeit von 1.702 ns bei 12.75 ns die Spitze des gesamten Kurve mit 1.557 V. Nach der Überschwungung stabilisierte sich die Impulsform ab ungefähr 20 ns auf der Top Line bei 1.486 V. Mit der in der Entwurfsregel vordefinierten Stoppzeit fällt sie nach 30 ns ab. Durch die Signalunterschwingung bleibt sie auf der Base Line bei 999.4 mV wieder konstant. Die Differenz zwischen High Pegel und Low Pegel beträgt 486.6 mV und liegt damit deutlich unter dem laut Hersteller-Datenblatt erlaubten Maximum von 600 mV.

Die Kurvendarstellung für zwei andere Eingänge, nämlich A39 auf dem Bottom Layer und B37 auf dem Top Layer beim Steckkarteverbinder U1 sieht fast identisch aus - mit minimaler Abweichung (siehe Abbildung 8-14). Z.B. betragen die Unterschwingungen 805.4 mV bei Pin A39, bei Pin B37 hingegen 825.9 mV. Die Abweichung beträgt also 20.5 mV. Bei Überschwüngen ist die Abweichung noch geringer, nämlich nur 4 mV.

Das dem Ausgangspin 92 und den Eingangspins 88, A40 und, B38 angeschlossene Netz NetIC1G1\_92, A\_40 ist das andere Mitglied des differentiellen LVDS Leitungspaares, das in den Entwurfs-Regeln mit Start Level High definiert wurde. Deswegen werden die Kurven in Abbildung 8-15 einfach umgekehrt dargestellt.

In einer Analyse des Übersprechens wurde ermittelt, wie groß der Störungseinfluss des Angreifer Netzes auf das Opfer Netz ist (siehe Kapitel 2 unter Abschnitt 2.6.5). Darum wurden das Netz A\_39 als Angreifer Netz und das Netz A\_36 als Opfer Netz im Signal Integrität Panel definiert. Bei der Betrachtung des dem Opfer Netz angeschlossenen FPGA Eingangspin 82 wird nach 32.489 ns die maximale Störung mit 13.453 mV erreicht, wie in Abbildung 8-16 dargestellt.

Außer der Reflexionsanalyse und Analyse des Übersprechens wurde für eine LVDS Leitung in diesem Experiment auch die Ankunftsreihfolge der Signale bei alle Eingangspins unter-

sucht. Für diese Untersuchung wurde ein gemeinsamer Spannungsmittelpunkt benötigt. Der sogenannte Mittelpunkt wurde durch folgende Formel ausgerechnet.

$$\text{Spannungsmittelpunkt} = \frac{\text{TopLine} + \text{BaseLine}}{2}$$

(Anmerkung: die Werte der Top Line bzw. Base Line aller Eingangspins unterschieden sich um 1 mV, Darum wurde die Abweichung einfach ignoriert.)

Der errechnete Spannungsmittelpunkt liegt bei ungefähr 1.2427 V. Durch den Einsatz einer Positionsmarke wurden die Signallaufzeiten für alle Eingänge angezeigt.

Bei Pin A39 erreicht der Impuls um 10.544 ns den Spannungsmittelpunkt 1.2427 V. Bei Pin B37 um 10.529 ns und bei FPGA Eingangspin 87 erst um 10.997 ns (siehe Abbildung 8-17). Im Vergleich zu den Eingängen beim Steckkarteverbinder hat sich die Signalübertragung zum Eingang des FPGAs ungefähr 0.4 ns verspätet. Der Grund dafür ist, dass der Weg vom FPGA Ausgang durch Loopback wieder zum Eingang deutlich länger ist als die Wege zu den beiden Eingangspins des Steckkarteverbinders. Die sortierte Ankunftsreihfolge der Signale lautet, zuerst Pin B37 auf dem Top Layer, dann Pin A39 auf dem Bottom Layer und zuletzt der FPGA Eingangspin 87.

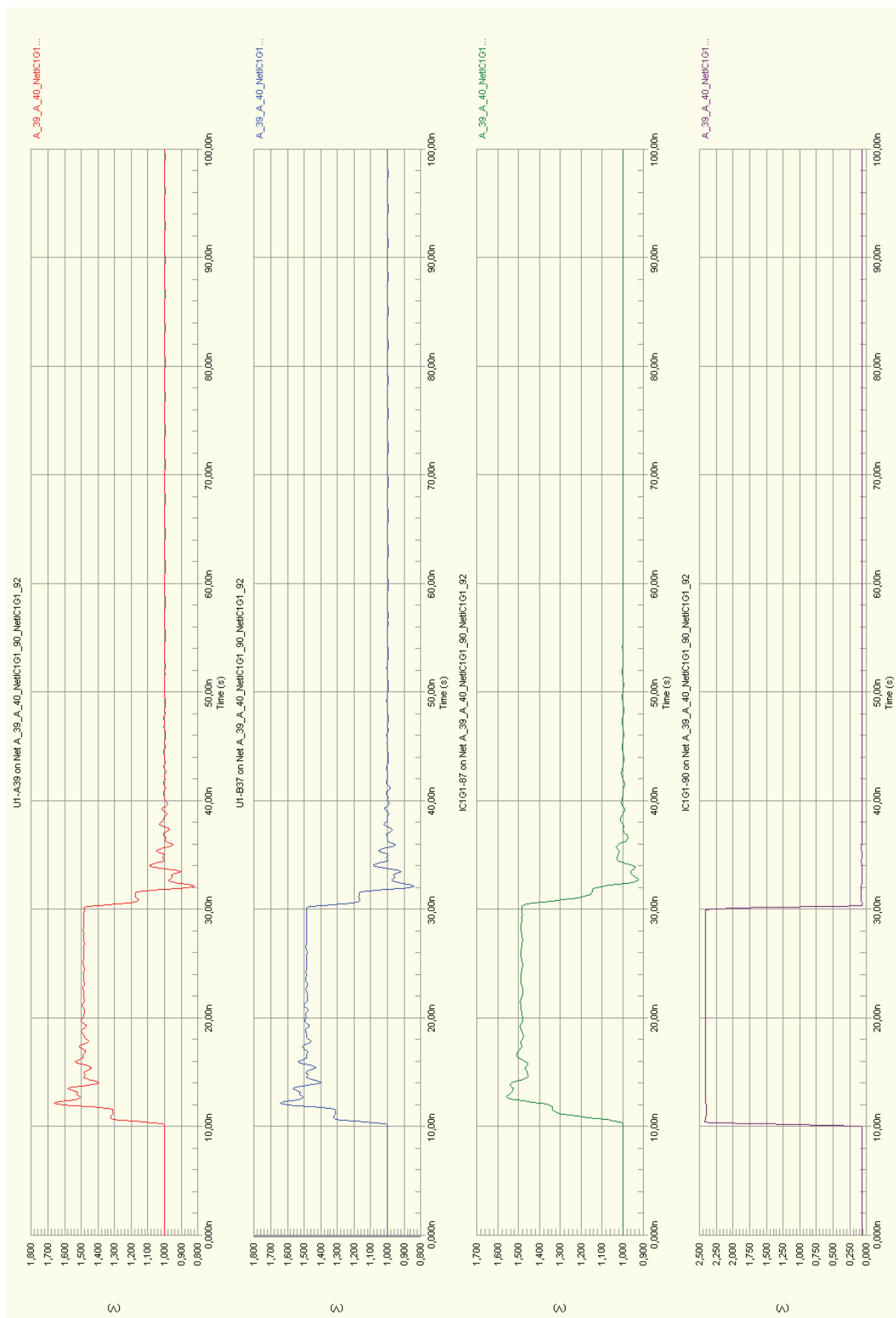
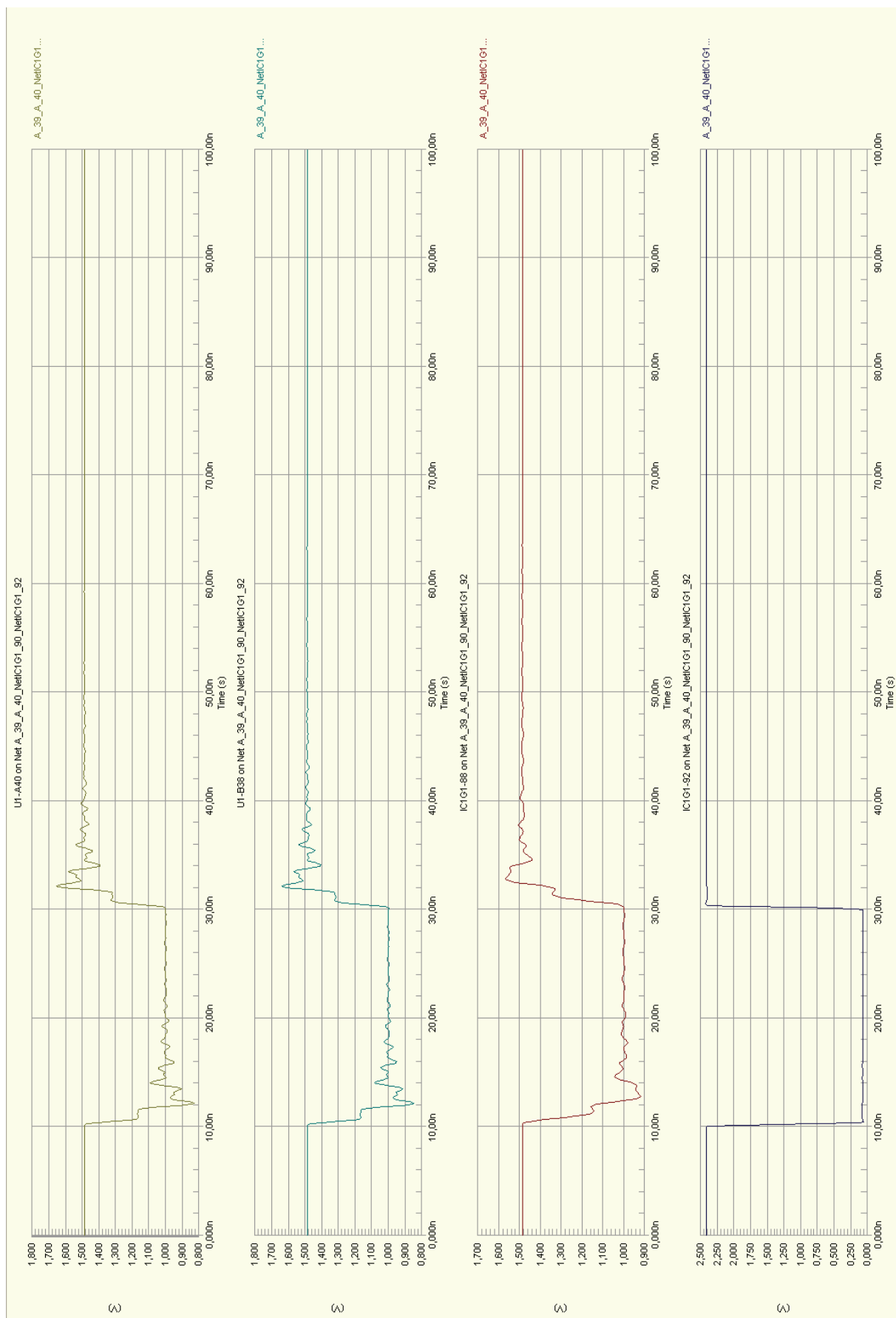
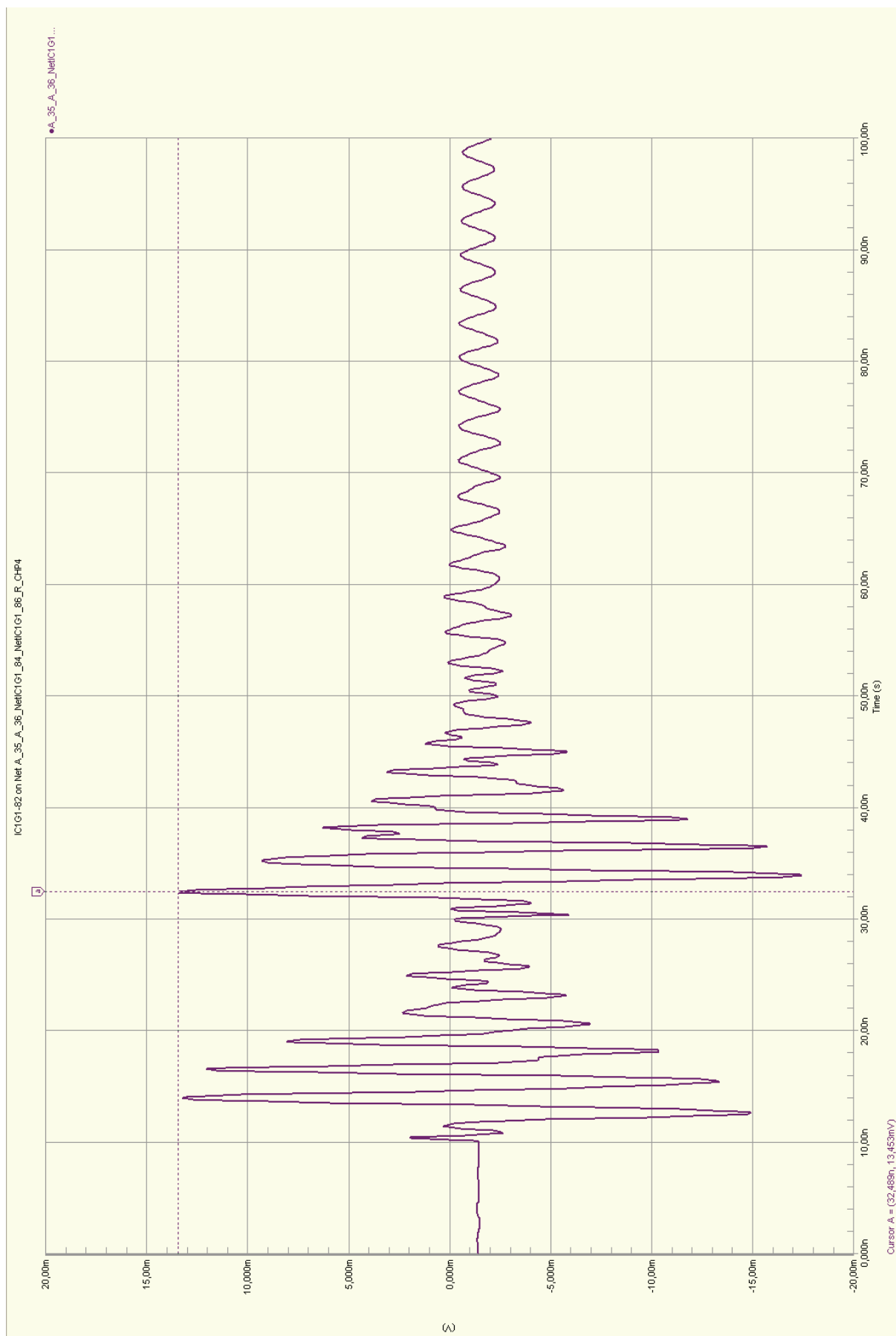


Abbildung 8-14: Ergebnis der Reflexionsanalyse mit High Pegel



**Abbildung 8-15: Ergebnis der Reflexionsanalyse mit Low Pegel**



**Abbildung 8-16: Analyseergebnis des Übersprechens**

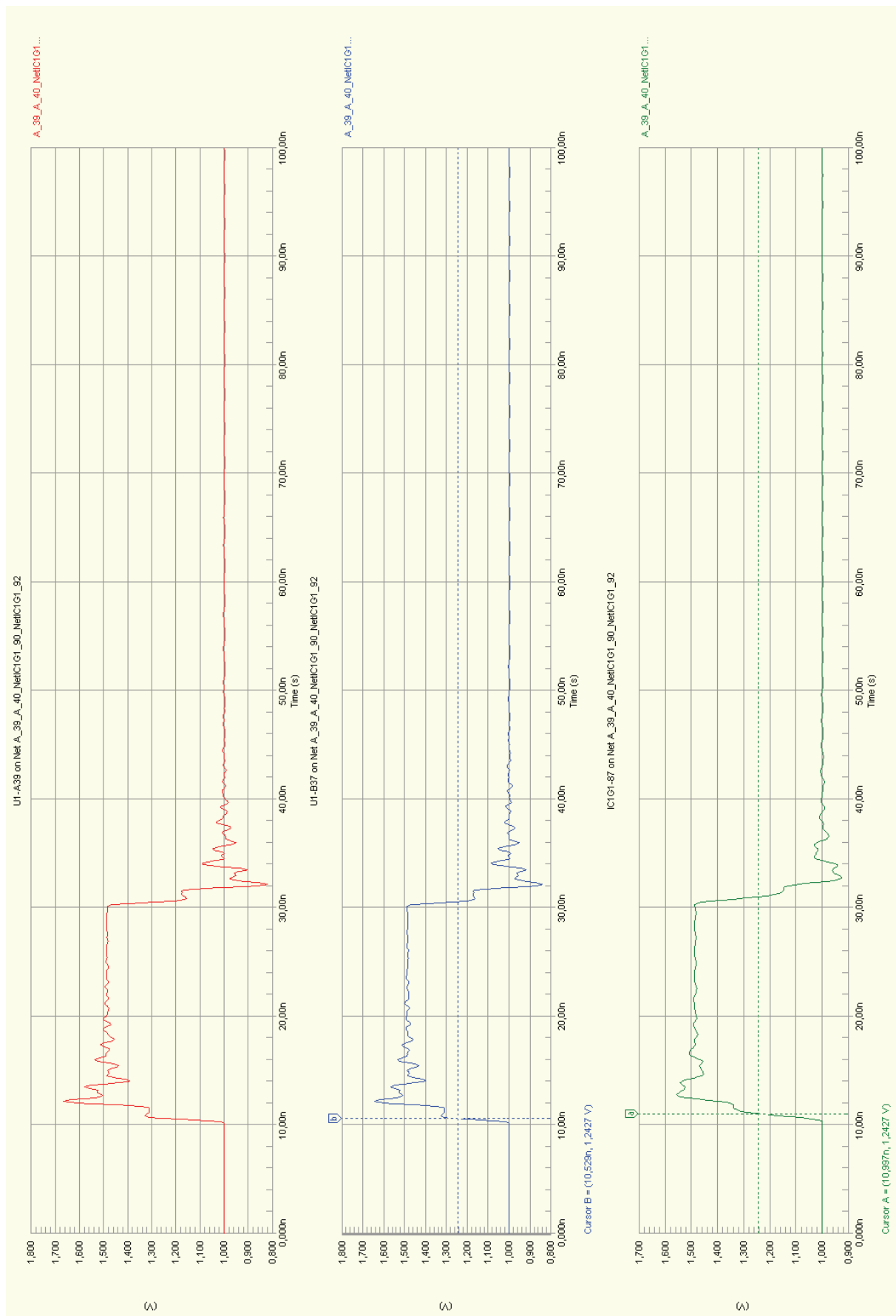


Abbildung 8-17: Ergebnis der Untersuchung der Signalankunftsreihfolge

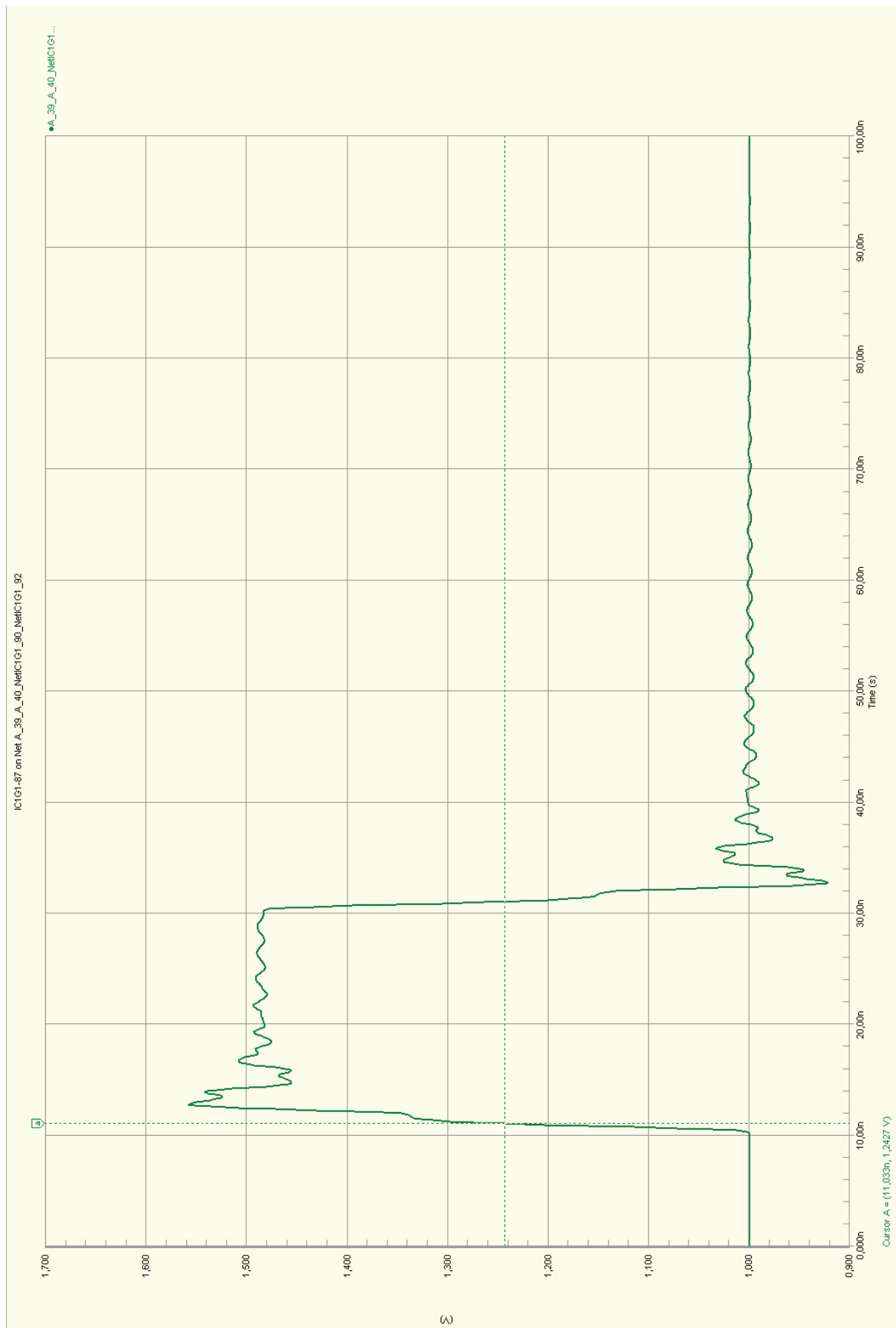


## **Das Ergebnis des zweiten Experimentes**

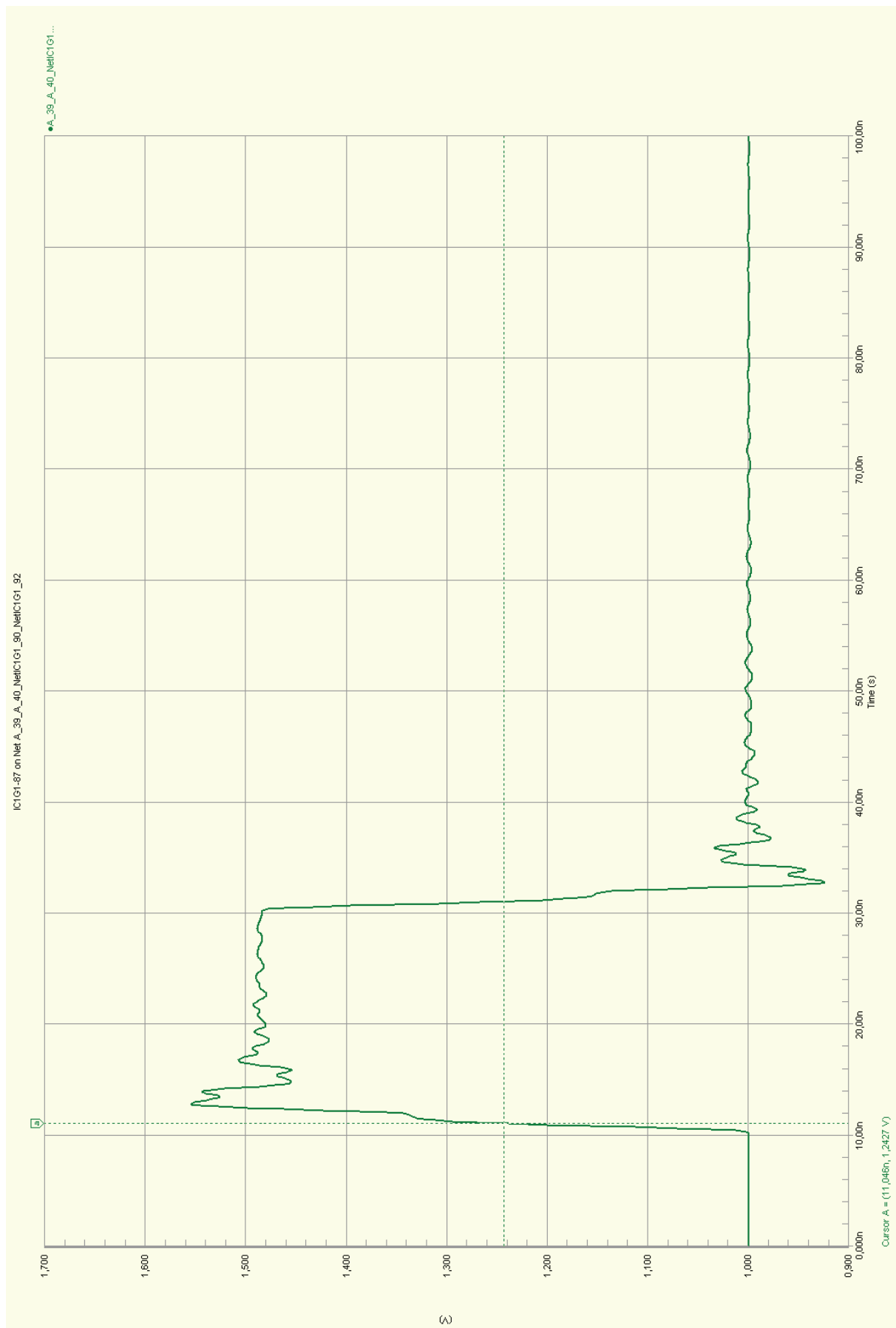
In diesem Experiment wurde, wie oben schon erwähnt, die durch die gebogene LVDS Leitung entstehende zeitliche Verzögerung untersucht. Dabei spielte auch der Spannungsmittelpunkt eine bedeutende Rolle. Der Wert wurde zu dem Zeitpunkt ausgelesen, wenn die Positionsmarke auf dem Spannungsmittelpunkt lag. Aus Abbildung 8-18 ist ersichtlich, dass das Signal bei Betrachtung des FPGA Eingangspin 87 ungefähr 11.033 ns gebraucht hat, um am Mittelpunkt anzulangen. Es ist 0.036 ns später als der erste Versuch.

## **Das Ergebnis des dritten Experimentes**

Das zentrale Augenmerk dieses Versuches war darauf gerichtet, Schlussfolgerung aus den Spannungsänderung zu ziehen, die entstehen, wenn die Impedanz sich durch die geringere Leitungsbereite erhöht und sich gleichzeitig der Abstand zwischen zwei LVDS Leitungen vergrößert. Das folgende Spannungsdiagramm (siehe Abbildung 8-19) des FPGA Eingangspin 87 vermittelt uns, dass sich das Signal zunächst um 0.049 ns verspätet. Bei der gesamten Impulsform ist offensichtlich keine Änderung zu sehen. Nur z.B. bei der ersten positiven Überschwingung hat das Signal einen Spannungswert von 1.5544 V erreicht, im Vergleich zum ersten Versuch ist das eine Verringerung um 2.9 mV. Beim zweiten positiven Überschwingen steigerte sich die Spannung aber um 1.4 mV. Beim Unterschwingen ist es genau umgekehrt. Die erste Unterschwingungsspannung zeigt einen minimalen Anstieg, die zweite Unterschwingungsspannung hingegen ist wiederum niedriger. Schließlich hat das Signal in diesem Versuch deutlich mehr Zeit gebraucht um die Spannung auf der Base Line bei 999.2 mV zu stabilisieren. Beim ersten Versuch ist die Spannung schon nach 80.00 ns auf der Base Line geblieben. Im Gegensatz dazu traten im dritten Experiment nach 80.00 ns noch einige Schwingungen auf.



**Abbildung 8-18: Analyseergebnis bei gebogene LVDS Leitung**



**Abbildung 8-19: Analyseergebnis bei erhöhter Impedanz**

## 9 Zusammenfassung und Ausblick

Auf der Basis meiner vorherigen Studienarbeit und mit der Referenz auf den bestehenden Prototyp in Eagle wurden die beiden Leiterplatten, nämlich Backplane und Basisboard, in dieser Arbeit mit der im Altium Designer integrierten Routing Funktion verdrahtet. Da sich die Autorouting Funktion als ungeeignet erwies, wurden die beiden Platinen von Hand geroutet. Die Leiterplattenverdrahtung ist eine ziemlich zeitaufwändige Arbeit. Trotz guter Vorbereitung können auch während des Routingprozess an bestimmten Stellen Probleme auftauchen, so z.B. bei der Angleichung der LVDS Leitungslängen auf dem Basisboard. Hierbei war eine Fehlerbehebung für die Fortsetzung dieser Arbeit notwendig. Anschließend wurde bei der Beseitigung der Fehler eine Entwurfsänderung für das Basisboard angefordert. Im Rahmen einer Betrachtung des Prototyps und der vorhandenen Diplomarbeit von Michael Müller wurden hinsichtlich der Leiterplattenbearbeitung einige Verbesserung eingearbeitet. Dies betrifft z. B. die fehlenden Steuerungsleitungen an der Speicherschnittstelle im Prototyp.

Im Vordergrund dieser Diplomarbeit stand die Recherchearbeit für die meisten bekannten Simulationsverfahren. Dabei ging es auch um die begrifflichen Beschreibungen der Theorie der Simulationsverfahren und einiger der von ihr verwendeten Algorithmen. Aufgrund der fehlenden Simulationsmodelldateien ist die Mix-Signal-Simulation im Moment für die beiden Platinen nicht möglich. Daher richteten sich die Experimente schwerpunktmäßig auf die Board-Level-Simulation, bzw. die Signalintegritätsanalyse.

Der zweite Teil dieser Diplomarbeit präsentiert drei Signalintegritätsexperimente auf dem Basisboard, die auch eigene Schwerpunkte hatten. Durch Einbau des Loopback Systems, das nicht für die Endproduktion vorgesehen war, musste auch hier zunächst eine unvermeidbare Entwurfsänderungen erfolgen. Um präzise Analyseergebnisse zu erzielen, mussten die Simulationswerkzeuge richtig konfiguriert werden und eine unumgängliche Überarbeitung des IBIS Modells erfolgen. An Hand des Altium Designer Demoprojektes Differentiale Pair wurden Erkenntnisse zum Spannungsbereich und zur Darstellung der Impulskurve gewonnen. Um die korrekte Einstellung der Simulationswerkzeuge herauszufinden und die Kurvendarstellung zu verbessern bedurfte es eines erheblichen Zeitaufwandes.

Das erste Experiment zeigte schließlich das erwartete Ergebnis der Signalreflexionsanalyse und Analyse des Übersprechens. Im Zuge der zweiten Experimentierphase wurde gezeigt, dass die Signalübertragung sich durch die Verlängerung des Leitungsweges verzögert. Das

das dritte Experiment verdeutlicht die Spannungsänderung auf dem FPGA Eingangspin im Bezug auf die erhöhte Impedanz.

Als zukünftige Arbeit wird das gesamte Kompaktsystem für noch mehr Simulation bereitgestellt. Die Simulation auf Board Ebene dient als technisches Hilfemittel, um überprüfen zu können, ob die anfertigte Leiterplatte sich wirklich wie erwartet verhalten wird. Das Analyseergebnis soll zu Verbesserungsvorschlägen und Anregungen für den Entwurf ermutigen. Daher ist diese Bearbeitungsphase generell unvermeidlich, bevor eine Leiterplatte in die Endproduktion geht. Außerdem bietet sie eine Möglichkeit, Simulationsmodelldateien für selbst hergestellte elektrische Komponenten zu kompilieren. Wenn alle Komponenten, die sich auf beiden Platinen befinden, auch eigene passende Simulationsmodelle besitzen, kann eine Mix-Signal-Simulation in Altium Designer sicher durchgeführt werden. Dabei kommt der Gleichstromanalyse, der Wechselstromanalyse, sowie dem Transientenanalysediagramm eine große Bedeutung zu.

## Literaturverzeichnis

- [Alt08a] Altium Wiki Englische Ausgabe. Altium Limited, 2008. – Online im Internet: <http://wiki.altium.com/display/ADOH/Home>, [Stand 10/2009]
- [Alt08b] Altera Corporation (Hrsg.): *Cyclone II Device Handbook*. Doc – ID: CII5V1 – 3.3. Silicon Valley: Altera Corporation, Februar 2008. – Online im Internet: [http://www.altera.com/literature/hb/cyc2/cyc2\\_cii5v1.pdf](http://www.altera.com/literature/hb/cyc2/cyc2_cii5v1.pdf), [Stand 01/2010]
- [Alt06] Altium Designer PCB Design Training Module. Altium Limited, 2006
- [Ana05] Analoge Simulation. Hannover: Electronic Design Automation – Entwurfsautomatisierung in Mikroelektronik der Universität Hannover, Oktober, 2005. – Online im Internet: [http://edascript.ims.uni-hannover.de/320b\\_Analogsimulation/pdf/AnalogeSimulation.pdf](http://edascript.ims.uni-hannover.de/320b_Analogsimulation/pdf/AnalogeSimulation.pdf), [Stand 10/2009]
- [Ant03] Dr. Anton, Mario ; Weber Jürgen: *Simulationsmethodik für integrierte Mixed-Signal Automotive Schaltkreise*. Heilbronn: Atmel Germany GmbH, März, 2003. – Online im Internet: <http://www.iss.tu-darmstadt.de/staff/jweber/Simulationsmethodik.pdf>, [Stand 10/2009]
- [Bli06] Blieferneicht, Dennis: *Mix – Mode Schaltungssimulation mit XSPICE – basierten Systemen – Eine Einführung*. Juli, 2006. – Online im Internet: <http://triphoeix.de/wiki/files/Spice-paper.pdf>, [Stand 10/2009]
- [Dai09] Dai, Rongtao: *Weiterentwicklung einer Hardware – Lösung für einen Multi – DSP – System*, TU Ilmenau, Studienjahresarbeit, Mai 2009
- [Die92] Dietmar, Ehrhardt ; Jürgen Schulte: *Simulation mit PSPICE – Eine Einführung in die analoge Schaltkreissimulation*. Braunschweig, Wiesbaden: Vieweg, 1992. –ISBN 3–528–04921–9
- [Dsi04] D.SignT GmbH (Hrsg.): *D.Module.C6713 User's – Guide*. Version 1.1. Kerken: D.SignT GmbH, September 2004. – Online im Internet: <http://www.dsignt.de/support/>, [Stand 10/2009]
- [Flc92] F.L. Cox ; W.B. Kuhn ; H.W. Li ; J.P. Murray ; S.D. Tynor ; M.J. Willis: *XSPICE Software Users Manual*. Atlanta, Georgia: Computer Science and In-

formation Technology Laborator –Georgia Tech Research Insititute, Dezember, 1992. – Online im Internet:

[http://users.ece.gatech.edu/~mrichard/Xspice/Xspice\\_Users\\_Manual.pdf](http://users.ece.gatech.edu/~mrichard/Xspice/Xspice_Users_Manual.pdf),

[Stand 10/2009]

- [Haa03] Haase, Joachim: *Regeln für die Erstellung von VHDL – AMS – Modellen*. Dresden: Fraunhofer – Institut Integrierte Schaltungen / Außenstelle Entwurfsautomatisierung, Februar, 2003. – Online im Internet: <http://www.bausch-gall.de/haase1.pdf>, [Stand 10/2009]

- [Haa99] Haase, Joachim ; Vermeiren, Wolfgang ; Clauß, Christoph ; Schwarz Peter: *Erste Erfahrungen mit der Simulation von Mixed – Signal – Schaltungen mit einem VHDL – AMS – Simulator*. Dresden: Fraunhofer – Institut für Integrierte Schaltungen, 1999. – Online im Internet: <http://www.eas.iis.fraunhofer.de/publications/papers/1999/027/paper.pdf>, [Stand 10/2009]

- [Har08] Hartl, Harald ; Krasser, Edwin ; Pribyl, Wolfgang ; Söser, Peter ; Winkler, Gunter: *Elektronische Schaltungstechnik mit Beispielen mit PSpice*. München: Pearson Studium, 2008. – ISBN 978–3–8273–7321–2

- [Hei09] Heinemann, Robert: *PSPICE Einführung in die Elektroniksimulation*. 6. Neu bearbeitet Auflage. München: Carl Hanser Verlage, 2009. – ISBN 978–3–446–41592–8

- [Her02] Herbert, Bernstein: *Simulationsverfahren in der Elektrotechnik – Grundlagen, Bauelemente, Grundsaltungen, Messtechnik mit analog – digitalem PC – Simulator*. Berlin: VDE VERLAG GMBH, 2002. – ISBN 3–8007–2596–7

- [Hir07] Hirsch, Martin: *Konzeption und Realisierung eines geschalteten Seriellen Kommunikationssystems für ein Mehr Prozessorsystem*, TU Ilmenau, Diplomarbeit, Oktober 2007

- [IBI02] IBIS White Paper – IBIS Model Process for High – Speed LVDS Interface Products. National Semiconductor Corp. Interface Products Group, 2002. – Online im Internet: [http://www.national.com/appinfo/lvds/files/wp\\_ibis\\_modeling.pdf](http://www.national.com/appinfo/lvds/files/wp_ibis_modeling.pdf), [Stand 01/2010]

- [Joe08] Joeres, Stefan: *Systemsimulationen zur Funktionalen Verifikation von HF- und Mixed – Signal – Schaltungen*. Aachen: Rheinisch – Westfälische Technische Hochschule, September 2008.
- [Kes06] Kesel, F. ; Bartholomä, R.: *Entwurf von digitalen Schaltungen und Systemen mit HDLs und FPGAs*. München: Oldenbourg Wissenschaftsverlag, 2006. – ISBN 978–3–486–57556–9
- [Kis03] Kiser, Andy: *Leiterplattendesign für Schnelle Signale*. Luzern: Fachstelle Elektronik, Hochschule Technik und Architektur Luzern, März 2003. – Online im Internet: <http://www.forelec.ch/forelec.ch/fichiers/HS-PCB-1.PDF>, [Stand 10/2009]
- [Mix05] Mixed – Signal – Simulation. Hannover: Electronic Design Automation – Entwurfsautomatisierung in Mikroelektronik der Universität Hannover, Oktober, 2005. – Online im Internet: [http://edascript.ims.uni-hannover.de/330b\\_MixedSignalSimulation/pdf/MixedSignalSimulation.pdf](http://edascript.ims.uni-hannover.de/330b_MixedSignalSimulation/pdf/MixedSignalSimulation.pdf), [Stand 10/2009]
- [Mül08] Müller, Michael: *Hardware und Software für Kommunikationsaufgaben in einem DSP – Multiprozessorsystem mit programmierbarer Logik*, TU-Ilmenau, Diplomarbeit, Mai 2008
- [Per02] Perry, Douglas L.: *VHDL – Programming by Example, Tool Usage for simulation, Synthesis, and Atspeed Debugging*. Fourth Edition. McGraw – Hill, 2002. – ISBN 0–07–140070–2
- [Sch05] Schlegel, Michael: *Mixed – Level – Simulation heterogener Systeme mit VHDL – AMS durch Multi – Architecture – Modellierung*. Chemnitz: Der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität Chemnitz, 2005. – Online im Internet: <http://archiv.tu-chemnitz.de/pub/2005/0190/data/Dissertation.pdf>, [Stand 10/2009]
- [Son07] Sonderforschungsbereich 622. Website, April 2007. – Online im Internet: <http://www4.tu-ilmenau.de/sf622/>, [Stand 10/2009]



- [Spi02] Spice interactive User Guide. University of California at Berkeley September, 2002. – Online im Internet: <http://bwrc.eecs.berkeley.edu/Classes/IcBook/SPICE/>, [Stand 10/2009]
- [Ste06] Steuerungs – und Feldbustechnik. Bremerhaven: Institut für Automatisierungs – und Elektrotechnik, Hochschule Bremerhaven, April, 2006. – Online im Internet: [http://www1.hs-bremerhaven.de/kmueller/s06ng/assets/stf\\_e1.pdf](http://www1.hs-bremerhaven.de/kmueller/s06ng/assets/stf_e1.pdf), [Stand 10,2009]
- [Suf08] Suffa, Christian: *Entwurf und Modellierung eines Kommunikationsprotokolles für ein Multi – DSP – System und dessen exemplarische Realisierung in FPGAs*, TU-Ilmenau, Diplomarbeit, Februar 2008
- [Xil08] Xilinx, Inc: Spartan – IIE FPGA Family Data Sheet. DS077 – 1. Version 2.3. San Jose: Xilinx, Inc, 18 June 2008. – Online im Internet: [http://www.xilinx.com/support/documentation/data\\_sheets/ds077.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds077.pdf), [Stand 01/2010]
- [Wik09a] Wikipedia – SPICE (Software). Wikimedia Foundation, San Francisco, 2009. – Online im Internet: [http://de.wikipedia.org/wiki/SPICE\\_\(Software\)](http://de.wikipedia.org/wiki/SPICE_(Software)), [Stand 10/2009]
- [Wik09b] Wikipedia – Boundary Scan Description Language. Wikimedia Foundation, San Francisco, 2009. – Online im Internet: <http://de.wikipedia.org/wiki/BSDL>, [Stand 01/2010]

## Abbildungsverzeichnis

Abbildung 2-1: Attribute zur Unterstützung des Wertaustausches zwischen dem digitalen und analogen Teil einer VHDL-AMS-Beschreibung.....	19
Abbildung 2-2: Simulationsanalysenarten .....	22
Abbildung 2-3: Parameteränderung für verschiedene Simulationstypen .....	23
Abbildung 2-4: Reflexionsanalyse (Darstellung aus [Alt08a]).....	29
Abbildung 2-5: Analyse des Übersprechens (Darstellung aus [Alt08a]).....	30
Abbildung 3-1: Überblick auf dem Kompaktsystem .....	33
Abbildung 4-1: LVDS-basiertes ringförmiges Kommunikationsweg .....	40
Abbildung 5-1: Ablauf für eine PCB Layout .....	43
Abbildung 6-1: Top-Level-Schematic (Darstellung aus [Alt08a]) .....	45
Abbildung 6-2: Festlegung für Mindestabstand.....	47
Abbildung 6-3: Festlegung Minimum solder mask sliver.....	48
Abbildung 6-4: Festlegung für Leitungsbereite .....	49
Abbildung 6-5: Festlegung Via Style.....	50
Abbildung 6-6: Design Rule Checking .....	52
Abbildung 6-7: Design Rule Typen .....	53
Abbildung 6-8: ERC Error Reporting .....	54
Abbildung 6-9: ERC Connection Matrix .....	55
Abbildung 6-10: Mindestabstand Regelverletzung.....	56
Abbildung 6-11: Footprints der problematischen Bus-Schnittstelle .....	57
Abbildung 6-12: 3D Body.....	59
Abbildung 6-13: Layer Stack Manager .....	61
Abbildung 6-14: Via Eigenschaft.....	62
Abbildung 7-1: Manuell Routing (oben) und Autorouting (unten) in Vergleich.....	65
Abbildung 7-2: Interactive Routing for Net .....	67
Abbildung 7-3: Netz Manager.....	69
Abbildung 7-4: High Speed Rule .....	70
Inventarisierungsnummer: 2010-02-24/019/IN01/2231	115

Abbildung 7-5: Interactive Length Tuning .....	71
Abbildung 7-6: Der Problembereich: Anpassung der Leitungen auf gleiche Länge .....	73
Abbildung 7-7: Umleitungen auf GND Layer .....	74
Abbildung 7-8: EMIF Steuersignal /AWE (oben), /ARE (unten).....	76
Abbildung 8-1: Einfachen LVDS Anwendung (Darstellung aus [Alt08b]).....	80
Abbildung 8-2: IBIS Modell Importieren .....	82
Abbildung 8-3: Komponente Pin Editor .....	84
Abbildung 8-4: FPGA Signal Manager.....	85
Abbildung 8-5: Anweisungen für Differentielle Paare .....	86
Abbildung 8-6: Signal Stimulus.....	88
Abbildung 8-7: Regeleinstellung für Versorgungsleitungen und Layer .....	89
Abbildung 8-8: Edit Buffer Option .....	90
Abbildung 8-9: Das Demoprojekt in Schaltplan(oben) und Layout(unten).....	92
Abbildung 8-10: Reflexionsanalyseergebnis des von Demoprojektes.....	94
Abbildung 8-11: Das Loopback System in Schaltplan(oben) und Layout(unten) .....	97
Abbildung 8-12: gebogene LVDS Leitung .....	98
Abbildung 8-13: Verdünnte LVDS Leitungen und Vergrößerter Abstand zwischen zwei LVDS Leitungen .....	99
Abbildung 8-14: Ergebnis der Reflexionsanalyse mit High Pegel .....	102
Abbildung 8-15: Ergebnis der Reflexionsanalyse mit Low Pegel .....	103
Abbildung 8-16: Analyseergebnis des Übersprechens.....	104
Abbildung 8-17: Ergebnis der Untersuchung der Signalankunftsreihfolge .....	105
Abbildung 8-18: Analyseergebnis bei gebogene LVDS Leitung.....	107
Abbildung 8-19: Analyseergebnis bei erhöhter Impedanz.....	108

## **Anhang**

Dieser Anhang beinhaltet verschiedene Schaltpläne und Layout Darstellungen in grafischer Form, sowie Auflistungen der Material, die während der PCB Layout Bearbeitung verwendet wurden. Wegen der ungünstigen Formatttransformation wurden diese Dokumente zum Einen im Verzeichnis „Anhang“ auf der Begleit-DVD gesichert und zum Anderen zusätzlich im Großformat ausgedruckt und der Arbeit angefügt.

## Verzeichnisstruktur der beigelegten DVD

Auf der beigelegten DVD befinden sich alle referenzierten Dokumentationen, Layout's und Schaltpläne der Platinen, nützliche Datenblätter, zwei elektronische Versionen dieser Diplomarbeit und zusätzlich verwendete Materialien.

Die DVD hat folgende Verzeichnisauflistung:

- **Anhang** enthält Layout, Schaltplan und Materialisten der Platinen in PDF Format
- **Autorouting** enthält die mit Altium Designer Autorrouting Funktion verdrahtete Backplane und Basisboard.
- **Bilder und Screenshots** enthält einige selbst gezeichnete Bilder und Flussdiagramme und alle ausgeschnittenen Bilder.
- **Datenblätter** enthält die verwendeten Herstellerdatenblätter.
- **Demoprojekt** enthält ein in Altium Desginer integriertes Signal Integrität Demoprojekt.
- **Diplomarbeit** enthält diese Diplomarbeit in elektronischer Doc und PDF Version.
- **Experimente** enthält die in der Arbeit erwähnten drei Versuche und die zugehörigen Analyseergebnisse.
- **IBIS Modelle** enthält das Originale IBIS Modell und das selbst geänderte IBIS Modell.
- **Neu Komponenten** enthält eine selbst gebaute Komponentenbibliothek.
- **Projekte** enthält die kompletten in Altium Designer entwickelten Schaltpläne und Layout's der Platinen, sowie das Kompaktsystem als Schaltplan.

## Thesen zur Diplomarbeit

1. Das eingebettete DSP-Kompaktsystem, das aus drei Basisboards und einer Backplane besteht, benötigt zusätzlich für die Zusammenarbeit drei Kommunikationssysteme. Ein Kommunikationssystem mit ringförmiger Topologie wurde von LVDS Technik ausgerüstet. Es bietet große Flexibilität und auch die hohen Übertragungsraten.
2. Es wurden einige Simulationsverfahren und zugehörige Grundbegriffe, sowie verwendete allgemeine Algorithmen vorgestellt und Anhand von Beispielen näher erläutert. Dabei wurde auch die Realisierung der Board-Level-Simulation und der FPGA-Level-Simulation in Altium Designer untersucht. Die Board-Level-Simulation, genauer Signalreflexionsanalyse und Analyse des Übersprechens ebnet den Weg für die spätere Implementierung auf unsere Experimente.
3. Vor dem Routingprozess der Leiterplatten wurden mehrere Vorbereitungsarbeiten gemacht. Im Wesentlichen zählt dazu eine Top-Level-Schematic zur hierarchischen Darstellung des gesamten Entwurfs, die richtige Festlegung der Entwurfsregeln, Bestimmung der Leitungsbereite, die Durchkontaktierung sowie das Festlegen eines Mindestabstands zwischen allen Leitungen und Komponenten usw.
4. Weiterhin wurden die Platinen nach vorgegebenen elektrischen Regeln überprüft, so konnten beispielweise Vernetzungsfehler im Schaltplan, und andere elektrische Fehler entdeckt werden.
5. Außer der elektrischen Regelprüfung wurden auch die beiden Platinen einer Entwurfsregelprüfung unterzogen um die Fehler auf dem Layout zu kontrollieren.
6. Das angestrebte Ziel dieser Arbeit war die Verdrahtung der beiden Platinen. Die Altium Designer eigene Interactive Length Tuning Funktion bietet eine komfortable Anpassung der LVDS Leitungslänge. Änderungen einer vorangegangenen Diplomarbeit wurden auch endgültig eingearbeitet.
7. Dieser Arbeit präsentiert Signalreflexionsanalyse und Analyse des Übersprechens unter drei Betrachtungsfällen.
8. Die skizzierten und ausgewerteten Analyseergebnisse helfen dabei die Spannungsverläufe auf den betrachteten Ein- bzw. Ausgangspins nachzuprüfen. Die hier praktisch

verwendeten Board-Level-Simulationsverfahren stellen einen Einstiegspunkt für zukünftige Arbeiten dar.

Ilmenau, 24. Februar 2010

(Ort, Datum)

.....

(Unterschrift)

## **Eidesstattliche Erklärung**

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich benannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht.

Ilmenau, 24. Februar 2010

(Ort, Datum)

.....

(Unterschrift)